Journal of Reliable Intelligent Environments (2023) 9:57–85
https://doi.org/10.1007/s40860-022-00192-3

ORIGINAL ARTICLE

Back propagation artificial neural network for diagnose of the heart disease

Jagmohan Kaur[1] · **Baljit S. Khehra**[2] · **Amarinder Singh**[3]

Received: 22 June 2022 / Accepted: 10 October 2022 / Published online: 12 November 2022

Abstract

Nowadays, coronary heart disease is one of the most fatal disease globally. Many researchers and medical technicians have developed and designed various computer-aided diagnosis systems using various machine learning models such as random forest, linear regression, rough set, naive bayes, artificial neural network, support vector machine, multivariate adaptive regression splines, K-nearest neighbor and decision tree to name a few. This era of digital health domain demands early prediction of heart disease which is the crucial need to control the global mortality rate of this particular disease. Commonly used methods for heart disease detection are clinical and expert dependant which makes them costly and inaccessible to the masses. In this paper, an efficient-cum-automated coronary heart disease diagnosis model is being proposed using multi-layered artificial neural network with back propagation algorithm. The proposed model compares the variation caused by different number of neurons used in the hidden layers for different transfer functions. The model has been implemented on Kaggle and Statlog heart disease dataset with thirteen clinical parameters. The experimental results attained an accuracy of 99.92% using six hidden layers with *tan-hyperbolic* transfer function. The results have been substantiated through statistical parameters and *k*-fold cross validation.

Keywords Heart disease · Digital health domain · Machine learning · Artificial neural network · Mean squared error

1 Introduction

Coronary heart disease (CHD) is a state of irregular cardiac rhythms arising due to blockage in the coronary arteries and blood vessels that obstructs the supply of oxygenated blood to the different parts of the body. The risk of heart disease increases due to non-detection at an early stage or ignorance of risk factors involved. About 795 thousand people in US [1] suffer from heart stroke and 137 thousand die annually. The most distressing thing is that maximum people die due to first heart stroke itself while the survivors still risk the chance of a second stroke within 5 years. Centers for Disease Control and Prevention (CDC) [1], US reported the death of an American every 40 s while American Heart Foundation (AHA) [2], US has identified that 25% of population die because of CHD.

Figure 1 shows the mortality rate of males and females announced by AHA in the year 2020. The mortality rate is highest for people with age more than 75 years but a significant figure is also seen for the age group of 25–34 years. Major factors causing CHD are physical inactivity, overweight, hyperglycemia, hypertension, life style and gene changes, wrong food and sleep habits.

The heart disease can be identified by typical symptoms that include chest pain/angima, swelling or numbness in legs and arms, breath shortness, pain in jaw, neck, throat and high heart rate. In the initial stage, the doctor/medical practitioner may prescribe drugs to control blood glucose, blood pressure or cholesterol before diagnosing actual blockage in the arteries. The standard modes of diagnosing the heart dis-

Baljit S. Khehra and Amarinder Singh are contributed equally to this work.

✉ Jagmohan Kaur
mohnideep1234@gmail.com

Baljit S. Khehra
baljitkhehra74@gmail.com

Amarinder Singh
amarinder77@gmail.com

1 IKG Punjab Technical University, Jalandhar, Punjab 144103, India

2 Computer Science and Engineering, BBSB Engineering College, Fatehgarh Sahib, Punjab 140407, India

3 Department of Applied Sciences, BBSB Engineering College, Fatehgarh Sahib, Punjab 140407, India

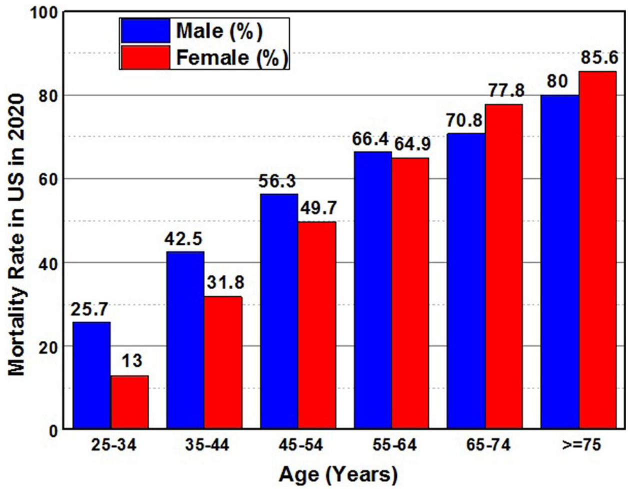


Fig. 1 US mortality rate in 2020

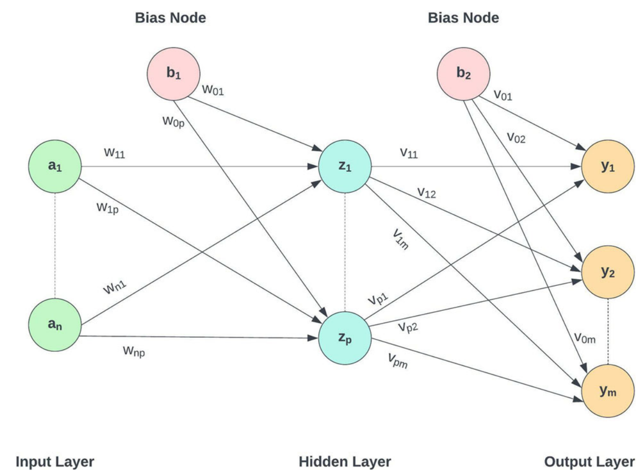


Fig. 2 The architecture of ANN

ease are blood tests, chest X-rays, electrocardiogram (ECG), cardiac computerized tomography (CCT) scan, magnetic resonance imaging (MRI), echocardiography, angiography and multiple-gated acquisition scanning (MUGA). Medical diagnosis depends largely upon expertise of doctors which may vary according to their personal experiences, past history of the patient and vague or incomplete medical data.

This has raised the demand for early detection of heart disease through diagnosis modeled on the modern technologies enabling the medical professionals, researchers and computer technologists to develop machine learning (ML) techniques for reliable, quick and digital diagnosis based on the fuzzy, indefinite and uncertain medical data. The use of smart devices such as smart watches, tablets, smart phones, internet of things (IoT) enabled gadgets, smart glass and portable health monitoring devices in day-to-day life can help to improve the disease detection at early stage.

IT professionals and researchers have introduced numerous data-mining classification techniques in the last few years for the prediction of heart disease. Support vector machine (SVM), Naive Bayes (NB), decision tree (DT), *K*-nearest neighborhood (KNN), logistic regression (LR) and artificial neural network (ANN) are the most popularly and commonly used machine learning (ML) algorithms for improving the accuracy of medical decision support system. However, automatic diagnostic system faces two major problems: overfitting and underfitting of the dataset which arises due to noisy and missing data, and division of training/testing datasets. The damage from these problems results in out of context trends in the final classification.

In order to remove these shortcomings an intelligent multi-layered ANN diagnosis system using back propagation is proposed in the study to enhance the accuracy of the model. Multi-layered ANN has the ability to solve non-linear and complex problems due to its high computational power. The emphasis on the choice of hidden layers and suitable combination of neurons has been made to avoid overfit and underfit of the data. The experiment work is carried over Kaggle and Statlog dataset. The performance on three transfer functions: *log-sigmoid, linear and tan-hyperbolic* in the hidden layers make it possible to analysis the best performance. The highlights of the paper are as follows:

- Demonstrates the latest advances for the design of heart disease diagnose model.
- Describes various combinations of neurons in different hidden layers.
- Explains achievements in the various phases of training/testing the system and analyzes related work using three transfer functions.
- Correlates mean squared error (MSE), accuracy and number of hidden layers.
- Evaluates the study using two data sets, Kaggle and Statlog, to formulate realistic model.

2 Study environment and background theories

ANN is one of the most famous ML adaptive model being used in the prediction of heart disease. ANN [3] or, simply, Neural Network is the simulation of decision making ability of human brain. Neural Network imitates the activity performed by neurons of brain.

A Neural Network, depicted in Fig. 2, mainly comprises three layers explained as under:

(i) The input layer receives the inputs and passes the same to second layer without processing. This layer has *n*

Table 1 Table of abbreviations of symptoms

S. no.	Symptoms	Units	Abbreviation	Range	Data type
1	Age		AG	30–86	Integer
2	Chest Pain (Typical Angina, Atypical Angina, Non- anginal pain, Asymptomatic)		CP	Yes, no	Catagorical
3	Cholesterol	mg/dL	CHL	125–200	Float
4	Resting Blood Pressure / Blood Pressure	mm/Hg	RBP/BP	75–120	Real
5	Elecrocardiogram / Resting Electrocardiogram	ms	ECG/ R ECG	120–200	Real
6	Maximum Heart Rate / Maximum Heart Rate Achieved during exercise	bpm	MHR / THALACH	Varies	Integer
7	Thallium Scan		THAL	Normal, Fixed Defect, Reversible Defect	Catagorical
8	Gender		GR	Male, Female	Catagorical
9	Old Peak (ST depression)		OP	6.2	Real
10	Slope of the Peak-Exercise ST-Segment		SLOPE	Upsloping, flat, downsloping	Catagorical
11	Fasting Blood Sugar	mg/dL	FBS	62–400	Real
12	Number of major vessels colored by fluoroscopy		CA	0–3	Integer
13	Exercise-induced Angina		EXANG	Yes, no	Catagorical

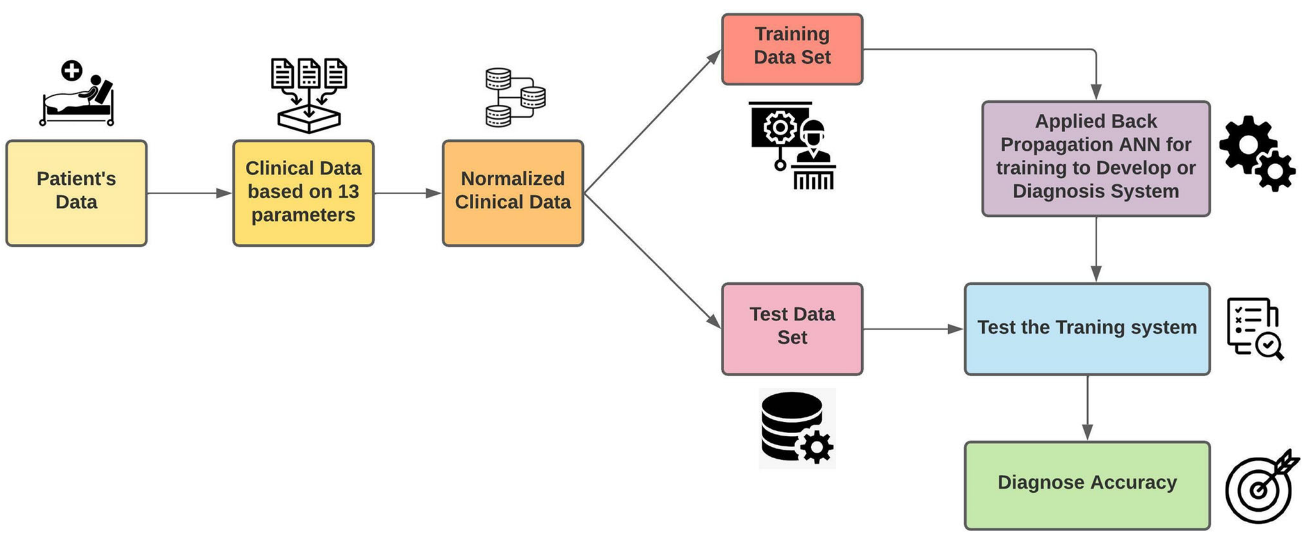


Fig. 3 The Block diagram of the proposed system

neurons $a_1, \ldots, a_n$ that are connected to every neuron of the second layer as explained ahead.

(ii) The second layer, known as hidden or processing layer, has p neurons $z_1, \ldots, z_p$ that are connected to synapses to input neurons where $w_{ij};\ 1 \le i \le n,\ 1 \le j \le p$ denote the weight of the arc between ith input neuron and jth hidden neuron. Thus, the input to each hidden neuron is the weighted sum of the outputs of the input neurons and second layer performs intermediary calculations using the summation and transfer function. All the hidden neurons are connected to each of the output neurons with weights $v_{ij};\ 1 \le i \le p,\ 1 \le j \le m$ for arc between ith hidden neuron and jth output neuron.

(iii) The output layer with m neurons, y_1, y_2 and y_m, receive inputs from the hidden layer neurons as sum of products and apply transfer function

Additionally there are two bias nodes, b_1 and b_2 that are connected to the hidden and output layers respectively with the weights $w_{01}, \ldots, w_{0p}$ and $v_{01}, \ldots, v_{0m}$. Bias performs as an interceptor in a linear equation. It adjusts the output alongwith the weighted sum of the inputs to the neurons,

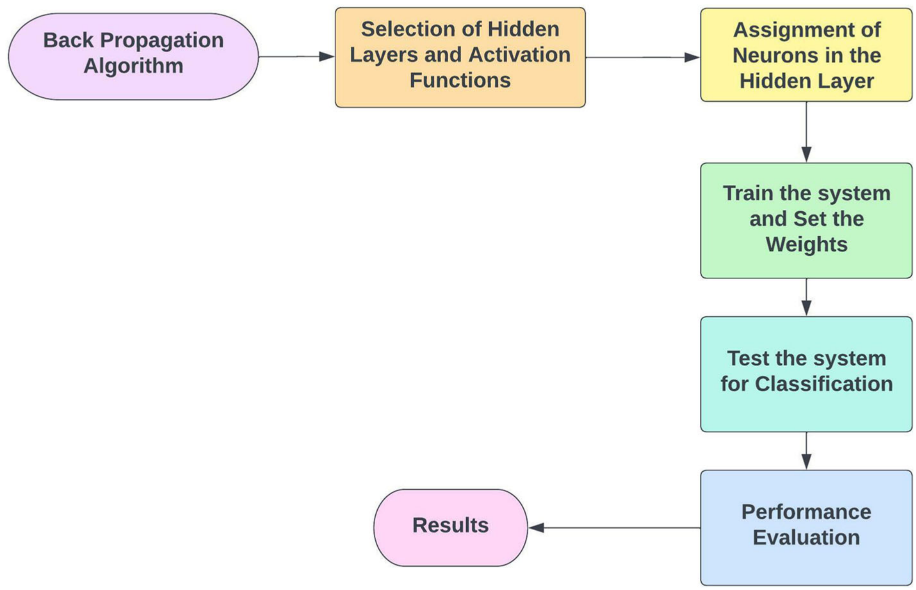


Fig. 4 Data-flow diagram of the proposed methodology

Table 2 Attainment of evaluation metrics in three layer ANN architecture

Combination of neurons	Iterations	CPU time (in s)	Best validation performance	Accuracy (%)		Validation (%)	Average
				Training	Testing		
(10, 9)	48	21	0.13251	91.20	88.96	94.94	91.41
(11, 10)	8	<1	0.179	78.77	65.22	70.82	75.60
(9, 8)	6	<1	0.16022	79.37	72.52	72.91	77.40
(10, 8)	72	22	0.15992	83.37	72.14	70.43	79.80
(11, 9)	41	2	0.14672	88.99	84.53	80.83	87.03
(9, 7)	14	1	0.17349	86.19	**89.89**	86.15	86.74
(12, 11)	27	2	0.15276	88.79	73.25	85.86	85.99
(12, 10)	15	1	0.13886	85.62	80.60	79.00	83.84
(8, 9)	188	8	0.11156	93.65	88.52	94.46	93.02
(7, 8)	266	150	0.14591	90.41	83.86	90.59	89.40
			Average	86.63	79.95	82.60	85.02
			SD	4.95	8.62	9.29	5.84

Bold value signifies the highest testing accuracy for distinct iterations

resulting in convergence of thereby determining the best fit for the given data.

Artificial Neural Network is applicable in various disciplines [4] including science, computing, engineering, agriculture, environmental, technology, mining, arts, climate, business, nanotechnology etc. In health-care and medicine [5], ANN is being applied extensively in general practice, internal medicine, invasive medicine, intensive care, anaesthesia and surgery. Apart from these, ANN is also being utilized in neuro [6], radiation medicine [7], forensic [8], dental science [9], dermatology [10], urology [6], ophthalmology [11], gynaecology [12], paediatry [13], gerontology [14], oncology [15], endocrinology [16] and cardiology [17]. The expanse of ANN is being leveraged to devise new solutions to the complex health care problems through the huge database of patients. It is not only an effective tool in classification of medical and health science diagnoses problems but beneficial for solving many other prime problems like prediction of signals/factors, signal enhancement and its identification. Back propagation is a supervised learning technique to efficiently train ANN. The aim of back propagation is to minimize the calculated error of the output using gradient descent or delta rule. It basically performs three phases of propagation: forward phase, backward phase and finally, substituting the obtained values together and computing the updated biases and weights. Fine-tuning of the biases and weights reduces error in the cost function and improve model confidence by generalizing. This technique follows the listed steps:

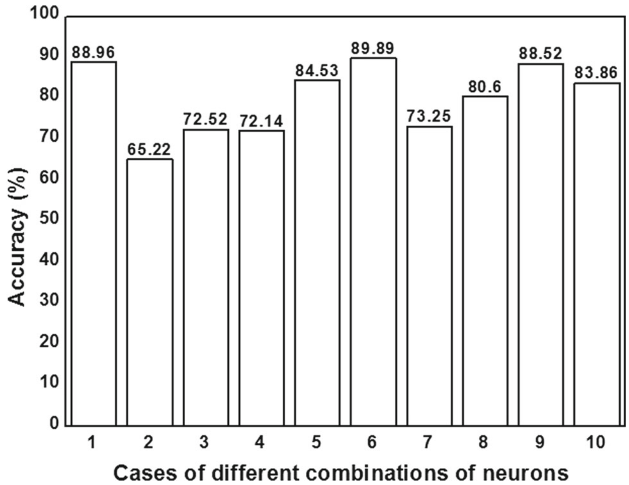


Fig. 5 Testing accuracy in three layer model

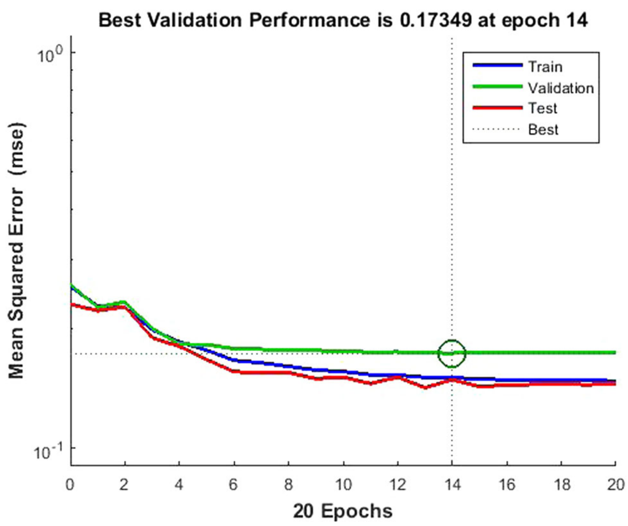


Fig. 6 Mean squared error of the best performance in three layer model

Step 1. Normalize of the input and output w.r.t their maximum values.
Step 2. Assign random weights to each neuron of the hidden and output layer of the network.
Step 3. Give an input unit a_i, $1 \le i \le n$ to each input neuron that passes it to the intermediate layer.
Step 4. Each hidden neuron z_j, $1 \le j \le p$ sums its weighted input signals through

$$z_{inj} = w_{oj} + n\sum_{i=1}^{n} a_i w_{ij} \tag{1}$$

Step 5. Compute the output signal and send it to all the units in the output layer.

$$z_j = f(z_{inj}),\ \text{where } f \text{is a transfer function} \tag{2}$$

Step 6. Each output neuron y_k, $1 \le k \le m$ sums its weighted input signals.

$$y_{ink} = v_{ok} + n\sum_{j=1}^{p} z_j v_{jk} \tag{3}$$

Step 7. Compute the output signal and send it to all the units in the output layer.

$$y_k = f(y_{ink}),\ \text{where } f \text{ is a transfer function} \tag{4}$$

Step 8. Each output unit receives a target pattern corresponding to the input training pattern and compute its error term

Table 3 Attainment of evaluation metrics in four layer ANN architecture

Combination of neurons	Iterations	CPU time (in s)	Best validation performance	Accuracy (%)		Validation (%)	Average
				Training	Testing		
(10, 9, 8)	48	18	0.14711	90.60	81.04	82.96	88.00
(11, 10, 9)	25	2	0.1573	90.20	**85.13**	80.43	87.99
(9, 8, 7)	42	29	0.18287	89.30	71.34	79.33	85.28
(10, 8, 7)	30	2	0.11647	84.34	84.41	90.87	85.35
(11, 9, 8)	30	3	0.15363	86.02	78.85	83.38	84.51
(10, 9, 7)	90	39	0.12791	92.62	84.30	90.66	91.07
(7, 8, 6)	34	9	0.14342	83.84	77.86	87.35	83.34
(9, 8, 6)	36	2	0.14628	87.46	81.73	85.37	86.22
(12, 11, 9)	12	1	0.16026	85.94	76.41	83.99	84.22
(10, 11, 9)	20	6	0.16659	88.37	77.04	77.54	85.06
			Average	87.87	79.81	84.19	86.10
			SD	2.86	4.35	4.50	2.31

Bold value signifies the highest testing accuracy for distinct iterations

$$\delta_k = (t_k - y_k)\acute{f}(y_{ink}) \tag{5}$$

Step 9. Calculate its weight correction term

$$\Delta v_{jk} = \alpha \delta_k z_j \tag{6}$$

where α is the learning rate.
Step 10. Calculate its bias correction term

$$\Delta v_{ok} = \alpha \delta_k \tag{7}$$

and sends δ_k to units in the layer below.
Step 11. Sum the delta inputs of each hidden neuron z_j

$$\delta_{inj} = \sum_{k=1}^{m} \delta_k v_{jk} \tag{8}$$

Step 12. Compute

$$\delta_j = \delta_{inj} f'(z_{inj}) \tag{9}$$

Step 13. Calculate its weight correction term

$$\Delta w_{ij} = \alpha \delta_j a_i \tag{10}$$

Step 14. Calculate its bias correction term

$$\Delta w_{oj} = \alpha \delta_j \tag{11}$$

Step 15. Update each weight of the network by

$$v_{jk}(\text{new}) = v_{jk}(old) + \Delta v_{jk} \tag{12}$$
$$w_{ij}(\text{new}) = w_{ij}(old) + \Delta w_{ij} \tag{13}$$

Step 16. Repeat steps 4–15 until the weights converge.

3 Related work

Researchers have worked on various ML models based on ANN or hybridized with ANN, convolutional neural network (CNN), recurrent neural network (RNN), radial Basis function (RBF), multi-layered perceptron neural network (MLPNN), back propagation neural network (BPNN), recurrent fuzzy neural network (RFNN) and long short term memory (LSTMNN) for automatic heart disease diagnosis. They have used several ensemble techniques to enhance the accuracy rate and analyzed the performance of the models through the MSE, receiver operator characteristic (ROC) curve, area under curve (AUC) and F1-score. Some of the related articles from 2010 to 2021 have been reviewed here.

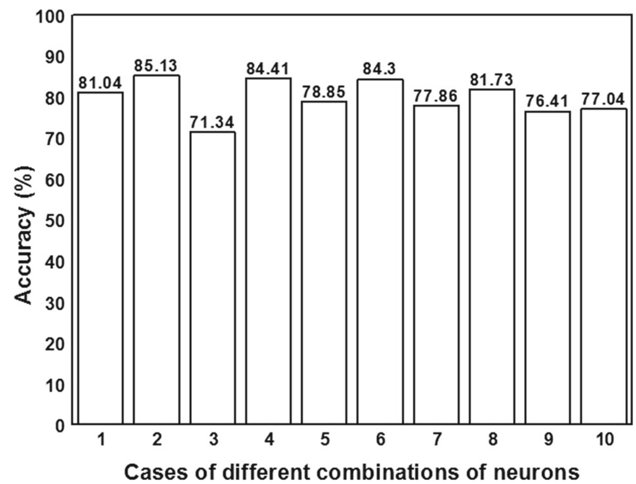


Fig. 7 Testing accuracy in four layer model

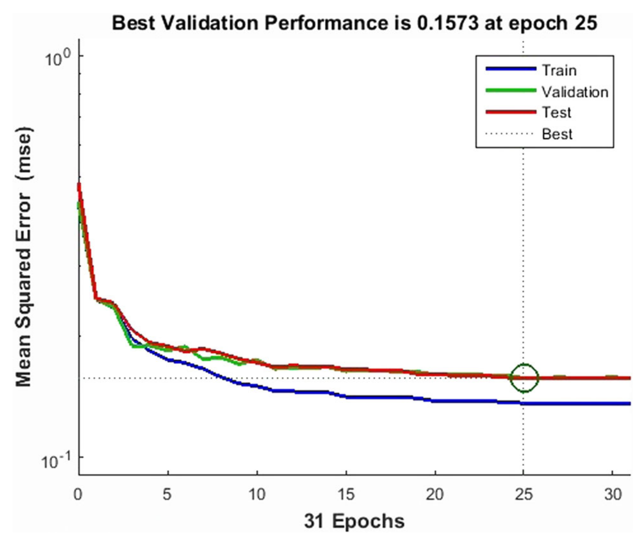


Fig. 8 Mean squared error of the best performance in four layer model

In 2019, Khourdifi and Bahaj [18] proposed classification and prediction of heart disease using ANN. The dataset consisting of 10 clinical attributes of 300 patients was taken from Sahara Hospital of Aurangabad. The neural network was trained and tested by RBF in MATLAB, and yielded 97% accuracy. Atkov et al. [19] created MLPNN in the year 2012. The paper presented a hybrid approach of genetic algorithm (GA) and ANN which are supposed to be suitable and supportive for non-linear and complex problems. It achieved an accuracy of 93% using genetic and non-genetic factors. The experimentation was done on 487 patients of Central hospital, Russia through NeuroSolutions tool of version 5.0.

An intelligent system to predict cardiovascular disease was explained in 2013 by Amma [20]. The study made an emphasis on amalgamation of multi-layered feed-forward ANN with back propagation and GA. A dataset of 303 patients from UCI, California, having thirteen attributes was used for training and testing the system, respectively. The

Table 4 Attainment of evaluation metrics in five layer ANN architecture

Combination of neurons	Iterations	CPU time (in s)	Best validation performance	Accuracy (%)		Validation (%)	Average
				Training	Testing		
(11, 10, 9, 8)	24	1	0.029799	98.57	**94.73**	94.13	97.31
(10, 9, 8, 7)	12	1	0.14453	86.76	80.91	81.36	85.00
(9, 8, 7, 6)	12	<1	0.18121	85.64	82.25	78.37	83.97
(10, 8, 6, 5)	11	3	0.12292	91.26	86.55	94.29	91.01
(10, 7, 6, 5)	17	<1	0.15662	92.16	80.48	80.01	88.59
(10, 8, 7, 6)	35	1	0.13457	87.80	83.24	81.61	86.28
(7, 9, 8, 6)	17	<1	0.1628	88.41	83.22	78.97	86.26
(12, 10, 8, 7)	38	1	0.13958	90.00	78.41	87.24	87.66
(9, 10, 7, 8)	12	1	0.18812	77.95	67.46	73.33	75.68
(10, 7, 6, 5)	19	<1	0.17656	83.87	77.92	75.71	81.70
			Average	88.24	81.52	82.50	86.35
			SD	5.46	6.90	7.19	5.69

Bold value signifies the highest testing accuracy for distinct iterations

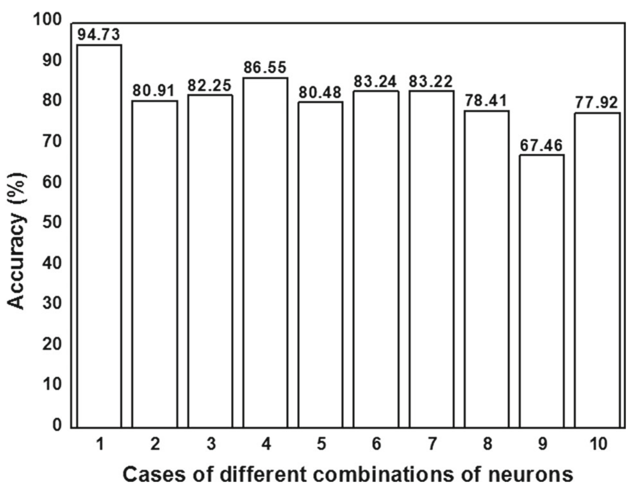


Fig. 9 Testing accuracy in five layer model

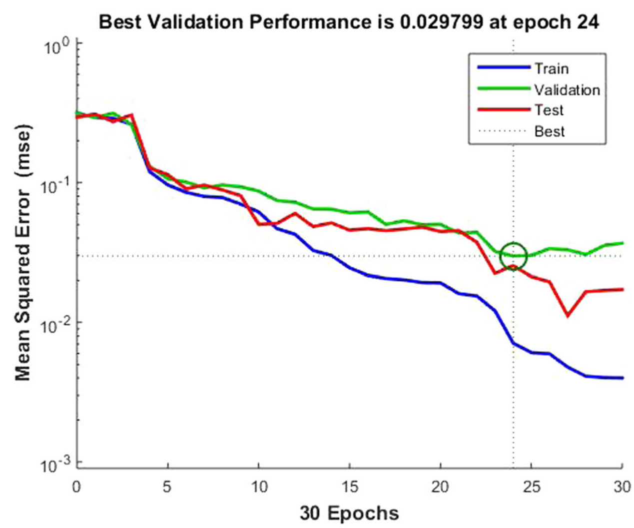


Fig. 10 Mean squared error of the best performance in five layer model

three layers of ANN consisted of thirteen, seven and one neuron respectively. The evaluation of weights of neural network yielded 94.17% accuracy.

Shao et al. [21] claimed classification of heart disease using various hybrid models of linear regression (LR), multivariate adaptive regression splines (MARS), ANN, rough set (RS) techniques in 2014. A dataset of 899 UCI patients, 60% for training and 40% for testing, with thirteen atributes was used. Using wald forward method in RESE software, single ANN technique yielded 76.79% accuracy with thirteen input nodes and hybrid MARS-ANN, LR-ANN and RS-ANN achieved 82.14%, 78.57% and 79.50% accuracy using six, twelve and ten input variables, respectively. Type-I and II error of best performance of MARS-ANN was recorded as 0.11 and 0.22 which were smaller than the models designed without ANN.

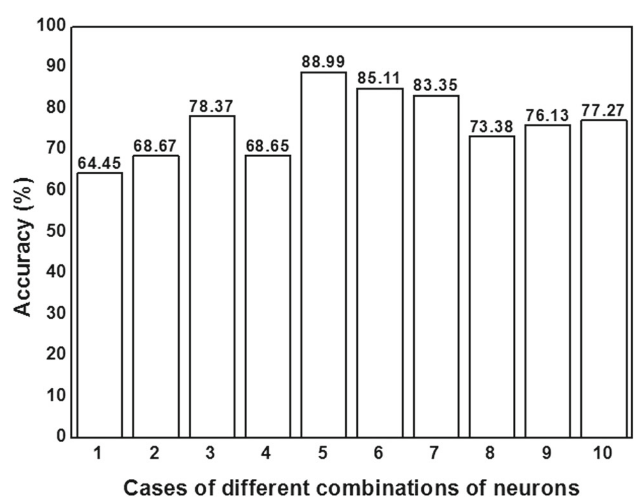


Fig. 11 Testing accuracy in six layer model

Table 5 Attainment of evaluation metrics in six layer ANN architecture

Combination of neurons	Iterations	CPU time (in s)	Best validation performance Performance	Accuracy (%) Training	Testing	Validation (%)	Average
(10, 9, 8, 6, 5)	22	1	0.17669	73.84	64.45	66.70	71.27
(9, 8, 6, 5, 4)	35	1	0.13234	87.19	68.67	86.13	84.23
(9, 8, 7, 6, 6)	26	1	0.12212	83.56	78.37	73.70	81.57
(9, 7, 6, 5, 4)	35	1	0.15112	82.05	68.65	79.13	79.57
(10, 8, 7, 5, 4)	22	<1	0.15285	86.91	**88.99**	80.56	86.29
(9, 8, 6, 5, 3)	58	1	0.13445	92.07	85.11	83.94	89.81
(11, 9, 8, 10, 5)	62	2	0.1343	87.56	83.35	85.70	86.65
(10, 11, 9, 8, 7)	21	1	0.12473	88.09	73.38	87.09	85.81
(12, 10, 9, 8, 7)	39	2	0.15811	86.75	76.13	85.54	84.94
(10, 9, 9, 8, 8)	50	2	0.14405	87.21	77.27	85.94	85.52
			Average	85.52	76.44	81.44	83.57
			SD	4.89	7.88	6.66	5.14

Bold value signifies the highest testing accuracy for distinct iterations

Feshki and Shijani [22] proposed selection of best features, using Particle Swarm Optimization (PSO), in the diagnosis system that are relevant with respect to time, cost and accuracy. Feature division of thirteen basic clinical attributes out of 76 features of Cleveland clinic foundation (CCF) dataset was made to design a diagnosis model at the minimum cost and time. BPNN within the PSO was implemented to extract eight features to test 303 patients to attain an accuracy of 91.94%.

A heart disease diagnosis model was presented by Uyar and Alhan [23] in 2017 utilizing GA-based RFNN involving thirteen input, seven hidden and one output neurons. The implementation on data of 297 CCF patients resulted in sensitivty (100%), specificity (95.54%), accuracy (97.78%), precision (96%), probable misclassification error (2.22), root mean square error (0.0222) and F1-score (0.9796).

An automated medical diagnosis system developed by Karayilan and Kiliç [24] in 2017 depicted BPNN using fourteen input parameters. The system included thirteen input neurons and two output neurons while in the hidden layer, three to twelve neurons were tested. The best accuracy of 95.55% was achieved at eight and eleven neurons in the hidden layer.

Gawande and Barhatte [25] in 2018 proposed a diagnostic model to predict heart disease using abnormalities in ECG waveforms. ECG signals of 340 patients of Massachusetts Institute of technology and Boston Israel Hospital (MIT-BIH) database were taken as input parameters. The data trained by CNN, consisting of total seven layers and sigmoid transfer function, revealed abnormalities of signals and illustrated the same in GUI form. The accuracy of the model was near 99%.

A smart clinical diagnostic model for heart was presented by Costa et al. [26] in 2019. The model utilized BLPNN to train 80% of the data taken from CCF and V. A. medical center comprising of thirteen clinical parameters. The testing of 20% data gave 90.74% accurate results using sigmoid transfer function with a learning rate and Nesterov's momentum of 0.28 and 0.15 respectively. The number of neurons in respective three layers were thirteen, six and one yielding AUC-score of 0.94, F1-score, precision and recall of 0.91.

Latha and Jeeva [27] came up with another heart risk diagnosis study combining the classification and ensemble techniques on a CCF data of patients in the age group of 29-79 segregated through NB, Bayes Net, C4.5 DT and MLPNN. Further, bagging, boosting and stacking ensemble techniques were employed to improve the accuracy. Finally, best accuracy of 80.53% and 84.16% was obtained using ANN and NB, respectively.

Muhammad et al. [28] demonstrated an adaptive computation model for an early early detection of heart disease. FCBF, maximum relevance minimum redundancy (MRMR), Least Absolute Shrinkage and Selection Operator (LASSO) and relief algorithms were applied to select features from a thirteen-parameters dataset taken from CCF and HIC. A 94.41% accurate result was achieved using ANN of thirteen, twenty and two neurons in three layers.

Recently, Shorewala [17] explored early prediction of heart disease employing various classification techniques including LR, SVM, RF, KNN, NB, MLPNN and DT. The model was tested on cardiovascular dataset of 70,000 patients wherein the feature selection was made using LASSO and bagging, boosting, and stacking ensemble techniques. DT achieved the highest accuracy of 74.8% and outperformed all other models.

In 2021, Shihab et al. [29] came up with a promising, supportive and feasible system revealing smart medical decisions

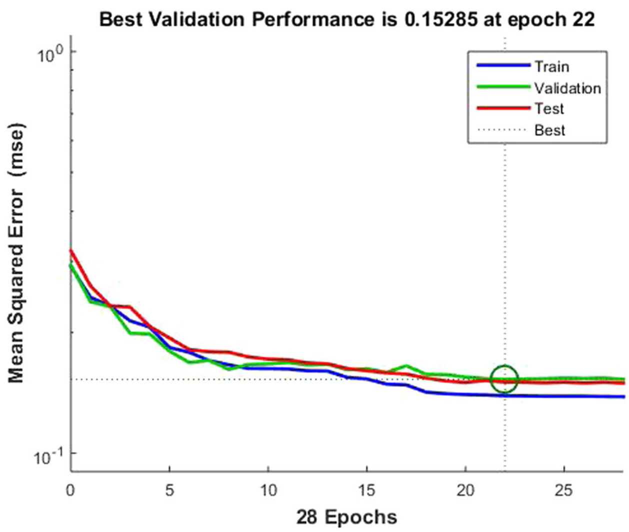


Fig. 12 Mean squared error of the best performance in six layer model

of heart failure through RNN and LSTM combining with IoT. Firstly, 15,000 ECG signals are collected from Py-serial library by Arduino. Then extraction of features from ECG signals was made and passed as input parameters for RNN and LSTM. The network was trained by back propagation algorithm using sigmoid transfer function. The system was tested on twenty people of Chattogram and yielded probable results.

Based on the critical analysis of the related work and comparative analysis of various NN models, the proposed study has employed BPNN due to its advantages. ANN based on back propagation can learn any non-linear function due to the application of all transfer functions. It enables the network to learn all the complex relationships between input and output parameters. It is fast, easy and simple to learn. In addition to this, there are no parameters to set other than the input number. This makes it possible to build a rigorous mathematical model. While CNN and RNN deals with major challenges like exploding gradient, class imbalance and overfitting. The application of these two NN require a huge amount of data and rate of convergence of these techniques is very slow. Furthermore, many iterations are required for MLPNN to learn to solve very simple logic problems.

4 Research methodology

The diagnosis of heart disease consists of input parameters that are the risk elements and are extracted from the dataset of thirteen clinical attributes, described in Table 1, of confirmed or suspected heart patients. The data is divided into two sets for training and testing and the classification of 'Presence' or 'Absence' of heart disease is made through the multi-layered ANN with back propagation algorithm as manifested in Fig. 3.

A diagnosis system (Fig. 4) is developed by training the network using different hidden layers and transfer functions, viz. Log-sigmoid Eq. (14) (*logsig*) [30], Linear (*purelin*) Eq. (15) and Tan-hyperbolic Eq. (16) (*tansig*) [31].

$$\phi(z) = \frac{1}{1 + \exp(-z)}, \quad z \in R \tag{14}$$

$$\phi(z) = \begin{cases} z & \text{if } z \geq 0 \\ -z & \text{if } z \leq 0 \end{cases} \tag{15}$$

$$\phi(z) = \frac{1 - \exp(-2z)}{1 + \exp(-2z)}, \quad z \in R \tag{16}$$

Finally, the trained system is tested and accuracy is evaluated at each layer using the said transfer functions. The MSE

Table 6 Attainment of evaluation metrics in seven layer ANN architecture

Combination of neurons	Iterations	CPU time (in s)	Best validation performance	Accuracy (%)		Validation (%)	Average
				Training	Testing		
(10, 9, 8, 6, 5, 4)	37	1	0.12977	86.75	80.31	90.05	86.19
(9, 8, 6, 5, 4, 7)	26	1	0.13855	82.24	74.57	84.31	81.48
(9, 8, 7, 6, 6, 5)	23	1	0.15002	85.17	66.41	70.38	79.99
(9, 7, 6, 5, 4, 6)	41	1	0.15193	85.13	84.44	87.48	85.39
(9, 8, 7, 5, 4, 7)	445	14	0.16695	85.94	77.73	72.88	82.62
(9, 8, 6, 5, 3, 5)	44	1	0.14263	87.75	72.55	90.28	85.84
(11, 9, 8, 10, 5, 7)	20	1	0.065774	94.40	87.85	85.98	92.15
(10, 11, 9, 8, 7, 10)	49	2	0.055556	100.00	**92.13**	88.95	97.13
(12, 10, 9, 8, 7, 11)	24	1	0.046741	99.86	91.53	90.58	97.21
(7, 9, 6, 9, 8, 5)	8	<1	0.17391	80.68	78.81	72.21	79.16
			Average	88.79	80.63	83.31	86.72
			SD	6.90	8.39	8.19	6.64

Bold value signifies the highest testing accuracy for distinct iterations

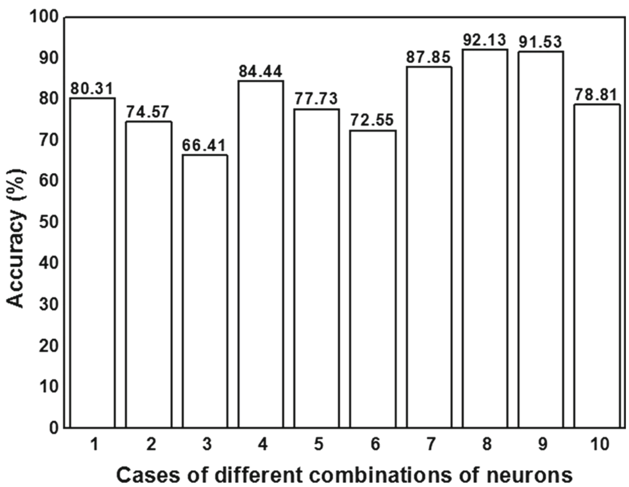


Fig. 13 Testing accuracy in seven layer model

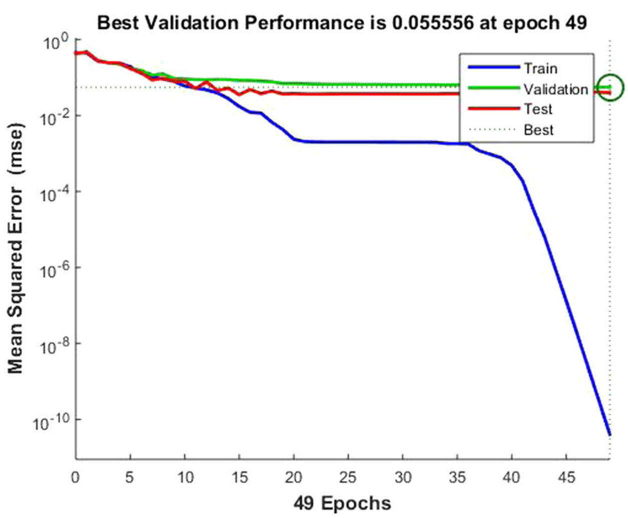


Fig. 14 Mean squared error of the best performance in seven layer model

Eq. (17) [32], is computed that shows the cumulative squared error between the actual and estimated value.

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{17}$$

where n is the number of data points, y_i and $\hat{y}_i$ represents observed and predicted values, respectively. The MSE of the best performance case is plotted to diagnose the system efficiently.

5 Experimental work

The implementation is carried out on Kaggle heart disease dataset of 1025 patients having thirteen clinical input parameters and two classes of output, 1 for 'Presence' and 0 for 'Absence'. After normalizing the data, it is segregated for training and testing processes. The observation of the experiments are recorded in 'nntool' toolbox of MATLAB R2015a.

5.1 Training of neural network via *log-sigmoid* transfer function

Neural network is trained by back propagation technique using hidden layers varying from two to seven and *log-sigmoid* transfer function. Average and variance of accuracy of each model is also computed to check the spread of data. The data includes iterations, time elapsed during training period and accuracy respectively along with validation. Table 2 present the data for neurons of two hidden layers. Figures 5 and 6 show the local maximum records of accuracy and validation respectively.

Table 7 Attainment of evaluation metrics in eight layer ANN architecture

Combination of neurons	Iterations	CPU time (in s)	Best validation performance	Accuracy (%)		Validation (%)	Average
				Training	Testing		
(10, 9, 8, 6, 5, 4, 3)	23	1	0.17658	84.97	77.45	70.02	81.63
(9, 8, 6, 5, 4, 7, 3)	21	1	0.15712	91.79	83.32	83.58	89.36
(9, 8, 7, 6, 6, 5, 3)	10	<1	0.19554	77.50	72.86	78.61	77.04
(9, 7, 6, 5, 4, 6, 2)	84	3	0.15283	79.55	77.78	77.09	78.87
(9, 8, 7, 5, 4, 7, 3)	45	1	0.13377	87.76	66.96	85.63	84.34
(9, 8, 6, 5, 3, 5, 2)	7	<1	0.14058	75.40	75.55	79.28	76.07
(11, 9, 8, 10, 5, 7, 4)	11	1	0.19371	83.97	67.51	71.66	79.72
(10, 11, 9, 8, 7, 10, 5)	19	1	0.16785	79.89	72.51	77.05	78.37
(12, 10, 9, 8, 7, 11, 5)	28	2	0.15505	89.94	**83.33**	78.10	87.16
(7, 9, 6, 9, 8, 5, 4)	23	1	0.17051	76.32	71.14	76.49	75.58
			Average	82.71	74.84	77.75	80.81
			SD	5.83	5.76	4.70	4.73

Bold value signifies the highest testing accuracy for distinct iterations

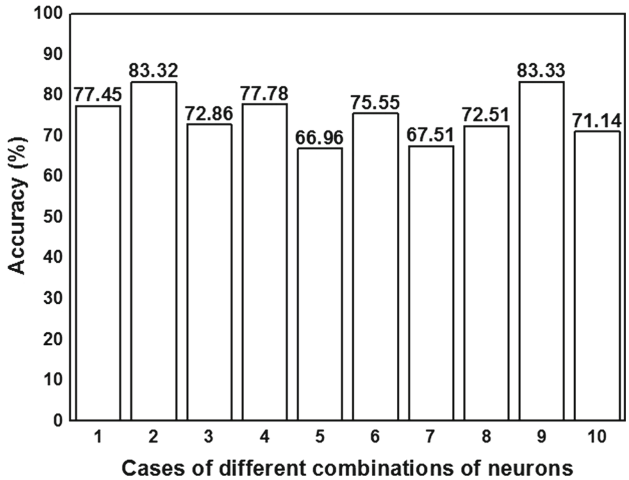


Fig. 15 Testing accuracy in eight layer model

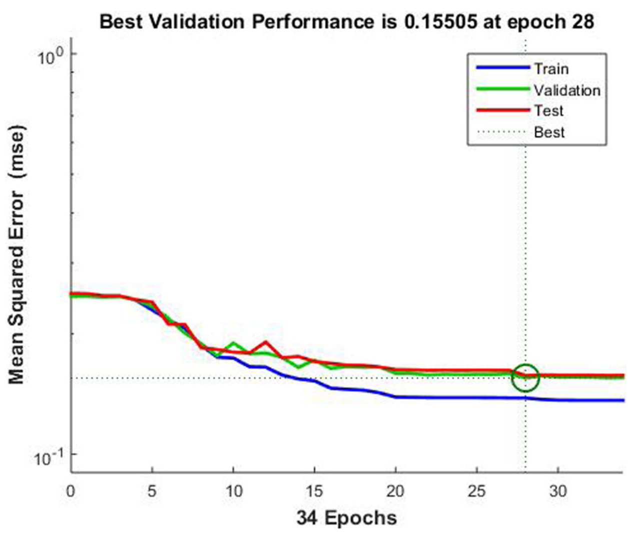


Fig. 16 Mean squared error of the best performance in eight layer model

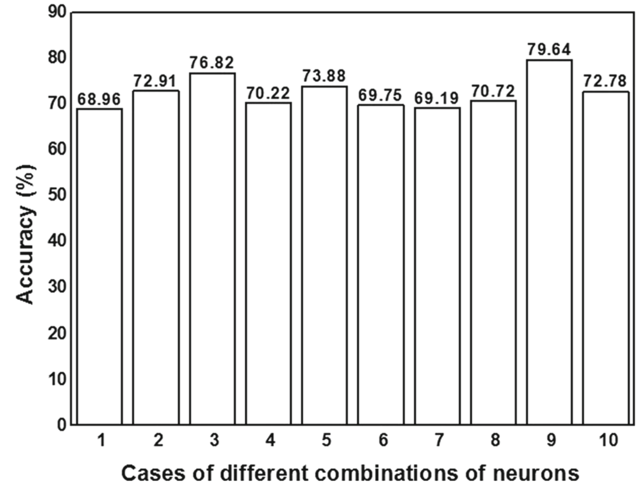


Fig. 17 Testing accuracy in three layer model

Evaluation metrics attained in four layer architecture are presented in Table 3. Accuracy bar graph and best validation performance achieved at the combination set of 11, 10 and 9 neurons are depicted in Figs. 7 and 8, respectively.

The best accuracy of *log-sigmoid* transfer function is obtained in five layer ANN architecture (Table 4) with training and testing accuracy are 98.57% and 94.73%, respectively and Figs. 9 and 10 represent accuracy of experimental tasks at distinct set of neurons and validation performance having MSE 0.029799 at 24th iteration.

The best performance of Table 5 is achieved at 28th iteration having 86.91 and 88.99% as training and testing accuracies. The local accuracy rate of five hidden layers is depicted in Fig. 14 and validation performance having MSE 0.15285 at 22th iteration is shown in Fig. 12.

Evaluation metrics attained in seven layer architecture are presented in Table 6. Accuracy bar graph and best validation

Table 8 Attainment of evaluation metrics in three layer ANN architecture

Combination of neurons	Iterations	CPU time (in s)	Best validation performance	Accuracy (%)		Validation (%)	Average
				Training	Testing		
(10, 9)	6	2	0.098742	73.34	68.96	78.20	73.44
(11, 10)	4	3	0.12526	73.97	72.91	70.30	73.53
(9, 8)	5	2	0.11138	72.43	76.82	74.25	73.46
(10, 8)	5	2	0.10502	73.49	70.22	76.39	73.45
(11, 9)	3	<1	0.12191	73.74	73.88	71.94	73.42
(9, 7)	4	1	0.11863	74.39	69.75	73.00	73.39
(12, 11)	2	3	0.15799	75.92	69.19	62.01	72.78
(12, 10)	2	2	0.12299	72.28	70.72	71.00	71.87
(8, 9)	6	<1	0.14284	73.42	**79.64**	65.48	73.26
(7, 8)	3	3	0.12082	73.75	72.78	72.58	73.30
			Average	73.67	72.49	71.52	73.19
			SD	1.02	3.51	4.81	0.51

Bold value signifies the highest testing accuracy for distinct iterations

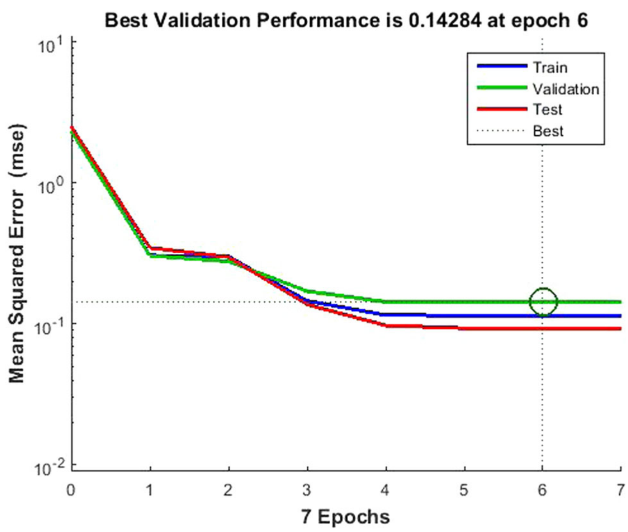


Fig. 18 Mean squared error of the best performance in three layer model

performance achieved at the combination set of 10, 11, 9, 8, 7 and 10 neurons are depicted in Figs. 13 and 14 respectively

The experimental records obtained in using eight hidden layers are shown in Table 7, and Figs. 15 and 16 illustrates graphical representation of accuracy of each record and validation execution corresponding to the maximum accuracy of 83.33% with the substitution of 12, 10, 9, 8, 7, 11 and 5 neurons in respective hidden layers.

5.2 Training of neural network via *linear* transfer function

Neural network is trained by back propagation technique using hidden layers varying from two to seven and *linear*

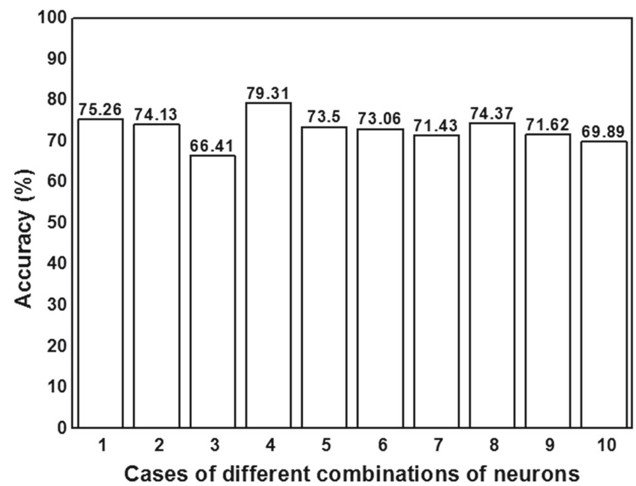


Fig. 19 Testing accuracy in four layer model

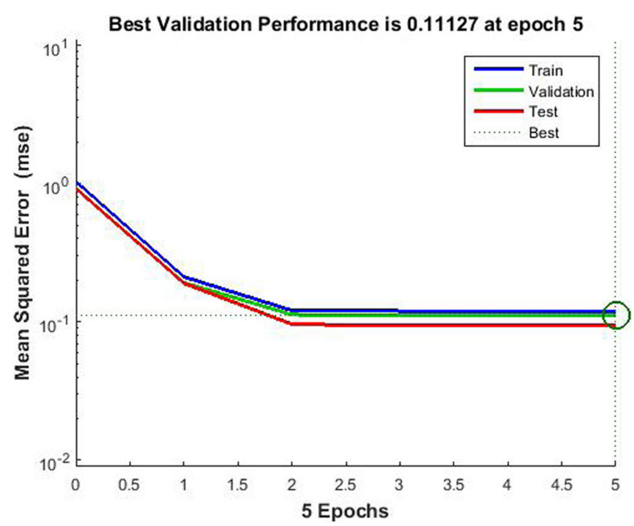


Fig. 20 Mean squared error of the best performance in four layer model

Table 9 Attainment of evaluation metrics in four layer ANN architecture

Combination of neurons	Iterations	CPU time (in s)	Best validation performance	Accuracy (%)		Validation (%)	Average
				Training	Testing		
(10, 9, 8)	6	2	0.11481	73.13	75.26	73.45	73.49
(11, 10, 9)	4	2	0.1041	72.83	74.13	76.55	73.57
(9, 8, 7)	3	2	0.091481	73.85	66.41	79.26	73.43
(10, 8, 7)	5	3	0.11127	72.05	**79.31**	75.31	73.52
(11, 9, 8)	4	4	0.10654	72.28	73.50	75.73	73.34
(10, 9, 7)	4	2	0.12966	74.09	73.06	69.86	73.12
(7, 8, 6)	6	3	0.092258	72.50	71.43	80.39	73.44
(9, 8, 6)	6	2	0.12976	74.29	74.37	69.84	73.53
(12, 11, 9)	4	4	0.094476	72.70	71.62	78.85	73.48
(10, 11, 9)	3	3	0.1547	76.48	69.89	64.23	73.56
			Average	73.42	72.90	74.35	73.45
			SD	1.32	3.43	5.08	0.13

Bold value signifies the highest testing accuracy for distinct iterations

Table 10 Attainment of evaluation metrics in five layer ANN architecture

Combination of neurons	Iterations	CPU time (in s)	Best validation performance	Accuracy (%)		Validation (%)	Average
				Training	Testing		
(11, 10, 9, 8)	4	<1	0.095794	72.57	73.34	78.73	73.60
(10, 9, 8, 7)	3	1	0.12445	73.36	75.88	72.02	73.39
(9, 8, 7, 6)	4	<1	0.11719	75.10	66.37	73.01	73.48
(10, 8, 6, 5)	3	<1	0.13345	74.64	72.51	68.11	73.35
(10, 7, 6, 5)	4	<1	0.12868	73.29	77.94	68.94	73.45
(10, 8, 7, 6)	6	<1	0.1458	76.55	65.60	65.12	73.32
(7, 9, 8, 6)	4	1	0.1068	72.25	76.68	75.96	73.49
(12, 10, 8, 7)	4	1	0.12698	72.43	**80.58**	70.27	73.33
(9, 10, 7, 8)	3	<1	0.13557	74.51	74.42	67.68	73.51
(10, 7, 6, 5)	6	<1	0.10108	73.21	70.83	77.28	73.49
			Average	73.79	73.42	71.71	73.44
			SD	1.38	4.81	4.50	0.09

Bold value signifies the highest testing accuracy for distinct iterations

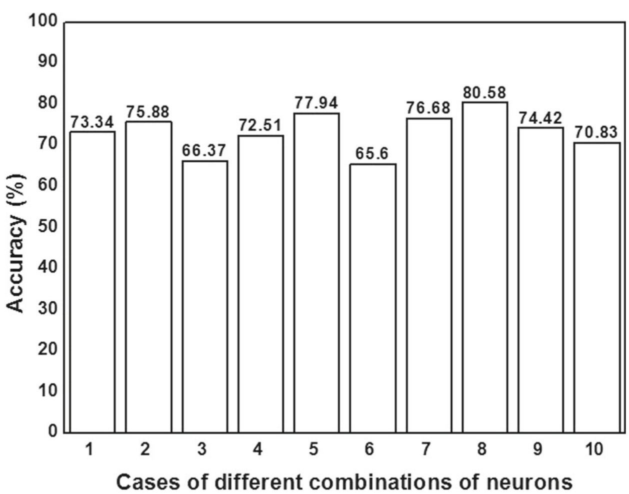


Fig. 21 Testing accuracy in five layer model

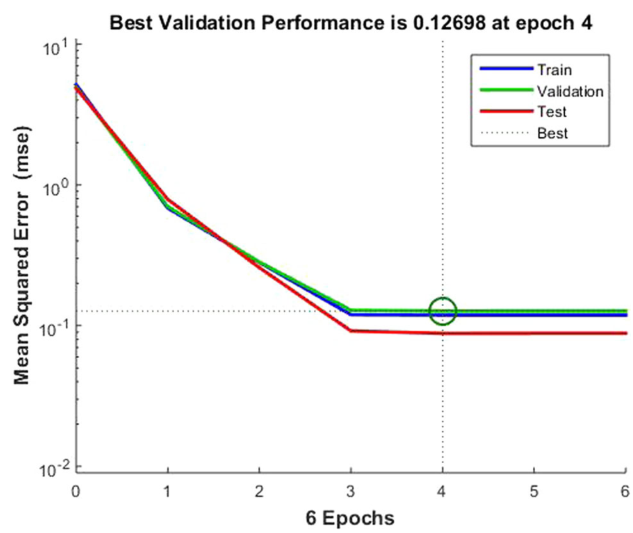


Fig. 22 Mean squared error of the best performance in five layer model

transfer function. Average and variance of accuracy of each model is also computed to check the spread of data. The data includes iterations, time elapsed during training period and accuracy respectively along with validation. Table 8 present the data for neurons of two hidden layers. Figures 17 and 18 show the local maximum records of accuracy and validation respectively.

Evaluation metrics attained in four layer architecture are presented in Table 9. Accuracy bar graph and best validation performance achieved at the combination set of 10, 8 and 7 neurons are depicted in Figs. 19 and 20 respectively.

The experimental records obtained in using four hidden layers are shown in Table 10, and Figs. 21 and 22 illustrates graphical representation of accuracy of each record and validation execution corresponding to the maximum accuracy of 80.58% with the substitution of 12, 10, 8 and 7 neurons in respective hidden layers.

The best performance of Table 11 is achieved at 7th iteration having 72.27 and 81.90% as training and testing accuracies. The local accuracy rate of five hidden layers is depicted in Fig. 23 and validation performance of blending 10, 9, 9, 8 and 8 neurons in respective layers with MSE 0.122 at 5th iteration is shown in Fig. 24.

The blending of 9, 8, 6, 5, 7, 3 and 5 neurons in respective six hidden layers of Table 12 marked 73.53% and 76.21% as training and testing accuracies. The illustration of accuracy records achieved at different combinations and best validation execution corresponding to the best accuracy is shown in Figs. 25 and 26.

The best accuracy of *linear* transfer function is obtained in the eight layer ANN architecture model shown in Table 13 where training and testing accuracy are 81.24% and 86.72%,

Table 11 Attainment of evaluation metrics in six layer ANN architecture

Combination of neurons	Iterations	CPU time (in s)	Best validation performance	Accuracy (%)		Validation (%)	Average
				Training	Testing		
(10, 9, 8, 6, 5)	3	<1	0.13293	74.99	69.59	66.88	73.06
(9, 8, 6, 5, 4)	5	<1	0.12203	74.81	68.25	71.96	73.40
(9, 8, 7, 6, 6)	4	<1	0.12378	74.72	70.17	71.25	73.58
(9, 7, 6, 5, 4)	5	<1	0.12087	72.69	77.85	71.19	73.40
(10, 8, 7, 5, 4)	5	<1	0.146	76.32	66.72	65.85	73.30
(9, 8, 6, 5, 3)	7	<1	0.13045	72.76	80.09	70.08	73.40
(11, 9, 8, 10, 5)	6	<1	0.12457	75.12	67.98	71.92	73.50
(10, 11, 9, 8, 7)	2	<1	0.13019	74.10	74.18	69.68	73.43
(12, 10, 9, 8, 7)	4	<1	0.12618	73.88	71.54	70.63	73.07
(10, 9, 9, 8, 8)	5	<1	0.122	72.27	**81.90**	71.43	73.50
			Average	74.17	72.83	70.09	73.36
			SD	1.28	5.41	2.11	0.17

Bold value signifies the highest testing accuracy for distinct iterations

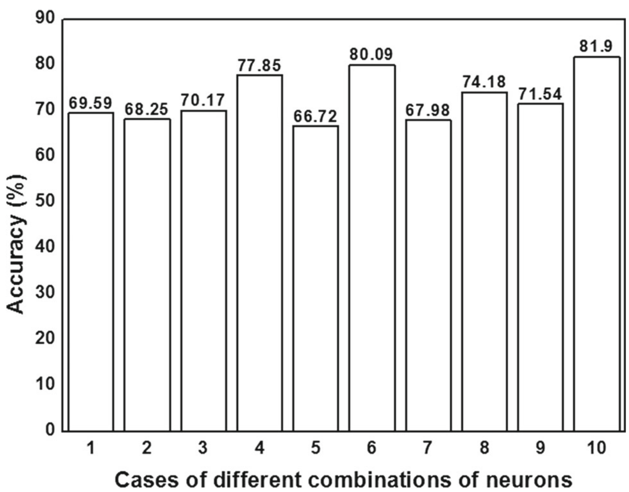


Fig. 23 Testing accuracy in six layer model

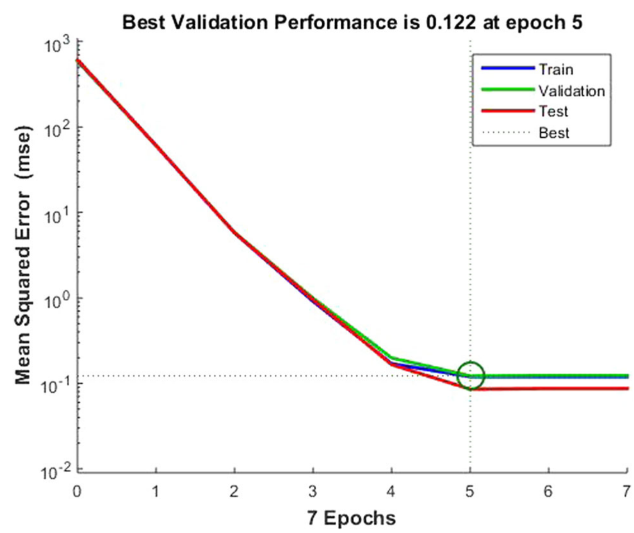


Fig. 24 Mean squared error of the best performance in six layer model

respectively. Figures 27 and 28 represent accuracy of experimental tasks at distinct set of neurons and validation performance having MSE 0.083134 at 14th iteration.

5.3 Training of neural network via *tan-hyperbolic* transfer function

Neural network is trained by back propagation technique using hidden layers varying from two to seven and *tan-hyperbolic* transfer function. Average and variance of accuracy of each model is also computed to check the spread of data. The data includes iterations, time elapsed during training period and accuracy respectively along with validation. Table 14 present the data for neurons of two hidden layers. Figures 29 and 30 show the local maximum records of accuracy and validation respectively.

The best validation performance for which the testing accuracy is the highest is presented in figs.
The best achievement of Table 14 is presented at 23rd iteration with 99.20% and 90.19% training and testing accuracy with best validation performance 0.040924 at 21st iteration shown in Fig. 30.

Evaluation metrics attained in four layer architecture are presented in Table 15. Accuracy bar graph and best validation performance achieved at the combination set of 12, 11 and 9 neurons are depicted in Figs. 31 and 32, respectively.

The experimental records obtained in using four hidden layers are shown in Table 16, and Figs. 33 and 34 illustrates graphical representation of accuracy of each record and validation execution corresponding to the maximum accuracy

Table 12 Attainment of evaluation metrics in seven layer ANN architecture

Combination of neurons	Iterations	CPU time (in s)	Best validation performance Performance	Accuracy (%) Training	Testing	Validation (%)	Average
(10, 9, 8, 6, 5, 4)	17	<1	0.12608	73.73	75.05	70.57	73.39
(9, 8, 6, 5, 4, 7)	6	<1	0.10556	72.44	76.14	76.42	73.57
(9, 8, 7, 6, 6, 5)	5	<1	0.11672	73.98	70.71	73.19	73.46
(9, 7, 6, 5, 4, 6)	6	<1	0.10142	73.07	69.96	77.71	73.33
(9, 8, 7, 5, 4, 7)	7	<1	0.10819	74.67	66.64	75.44	73.49
(9, 8, 6, 5, 3, 5)	6	<1	0.12426	73.53	**76.21**	70.59	73.42
(11, 9, 8, 10, 5, 7)	4	<1	0.10254	73.86	66.20	76.36	73.04
(10, 11, 9, 8, 7, 10)	6	<1	0.12637	73.87	75.45	71.50	73.47
(12, 10, 9, 8, 7, 11)	8	<1	0.12248	73.90	72.44	72.08	73.39
(7, 9, 6, 9, 8, 5)	7	<1	0.13058	75.26	66.61	69.56	73.29
			Average	73.83	71.54	73.34	73.38
			SD	0.77	4.10	2.92	0.15

Bold value signifies the highest testing accuracy for distinct iterations

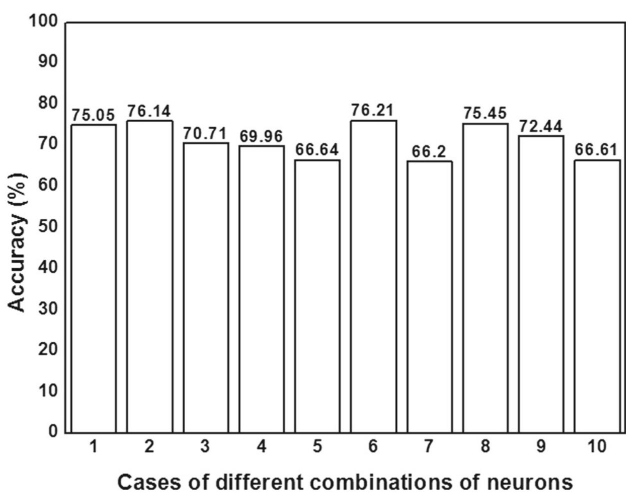


Fig. 25 Testing accuracy in seven layer model

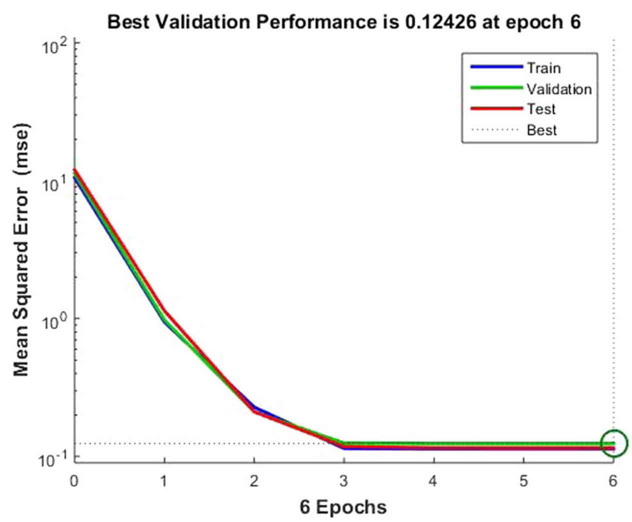


Fig. 26 Mean squared error of the best performance in seven layer model

of 94.76% with the substitution of 10, 8, 6 and 5 neurons in respective hidden layers.

The best performance of Table 17 is achieved at 50th iteration having 99.06 and 97.60% as training and testing accuracies. The local accuracy rate of five hidden layers is depicted in Fig. 16 and validation performance having MSE 0.021406 at 44th iteration is shown in Fig. 36.

The overall maximum accuracy of *tan-hyperbolic* transfer function is obtained in the seventh layer ANN architecture model shown in Table 18 where training and testing accuracy are 97.247% and 99.92%, respectively. Figures 37 and 18 represent accuracy of experimental tasks at distinct set of neurons and validation performance having MSE 0.027839 at 37th iteration.

The blending of 12, 10, 9, 8, 7, 11 and 5 neurons in respective seven hidden layers of Table 19 marked 93.78% and 95.79% as training and testing accuracies. The illustration of accuracy records achieved at different combinations and best validation execution corresponding to the best accuracy is shown in Figs. 39 and 40.

6 Comparative analysis of hidden layers of ANN and transfer functions

Accuracy of two, three, four, five, six and seven hidden layers with distinct pair of neurons has been computed for *log-sigmoid, linear and tan-hyperbolic* transfer functions of neural network. The computational and comparative analysis of the study is highlighted in the Figs. 41, 42, 43, 44, 45 and 46

Table 13 Attainment of evaluation metrics in eight layer ANN architecture

Combination of neurons	Iterations	CPU time (in s)	Best validation performance	Accuracy (%)		Validation (%)	Average
				Training	Testing		
(10, 9, 8, 6, 5, 4, 3)	6	<1	0.10161	79.61	74.65	77.22	78.52
(9, 8, 6, 5, 4, 7, 3)	7	<1	0.11485	80.51	78.26	73.81	79.19
(9, 8, 7, 6, 6, 5, 3)	8	<1	0.14644	78.45	83.02	64.75	77.18
(9, 7, 6, 5, 4, 6, 2)	8	<1	0.10593	81.30	73.00	76.10	79.27
(9, 8, 7, 5, 4, 7, 3)	5	1	0.12328	78.53	76.63	71.85	77.22
(9, 8, 6, 5, 3, 5, 2)	14	<1	0.083134	81.24	**86.72**	83.15	82.35
(11, 9, 8, 10, 5, 7, 4)	31	1	0.066913	82.18	84.24	85.76	83.06
(10, 11, 9, 8, 7, 10, 5)	22	01	0.10055	79.93	78.68	77.45	79.30
(12, 10, 9, 8, 7, 11, 5)	6	<1	0.38973	25.11	24.16	21.73	24.41
(7, 9, 6, 9, 8, 5, 4)	19	1	0.047585	83.77	73.367	90.05	83.19
			Average	75.06	73.27	72.19	74.37
			SD	17.63	17.87	19.14	17.70

Bold value signifies the highest testing accuracy for distinct iterations

revealing that at each layer *tan-hyperbolic* transfer function comes out with maximum accuracy of 90.19% (Fig. 41), 98.18% (Fig. 42) , 94.76% (Fig. 43), 97.6% (Fig. 44), 99.92% (Fig. 45) and 95.79% (Fig. 46) from three to eight layer ANN architecture respectively.

7 Results and discussion

Figure 47 plots the maximum accuracy corresponding to the number of hidden layers where each layer represents the best accuracy achieved for *log-sigmoid, linear and tan-hyperbolic* transfer functions of neural network.

It can be easily observed that the best performance is yielded corresponding to *tan-hyperbolic* transfer function in each layer and best accuracy of 99.92% is achieved in six layer model of ANN.

The performance of the proposed diagnosis system has been evaluated through some performance metrics and also validated through the k-fold method. Further more, the results of the proposed system have been compared with the other models and datasets.

7.1 Performance metrics

The confusion matrix [33] and the performances of the evaluation metrics [34] are tabulated in Tables 20 and 21 respectively with AUC 0.90 as plotted in Fig. 48.

7.2 Cross-validation *k*-fold method

Cross-validation is one of the best and popularly used to test the performance of the results that randomly partitions a dataset into k-subsets where $(k - 1)$ subsets are used for

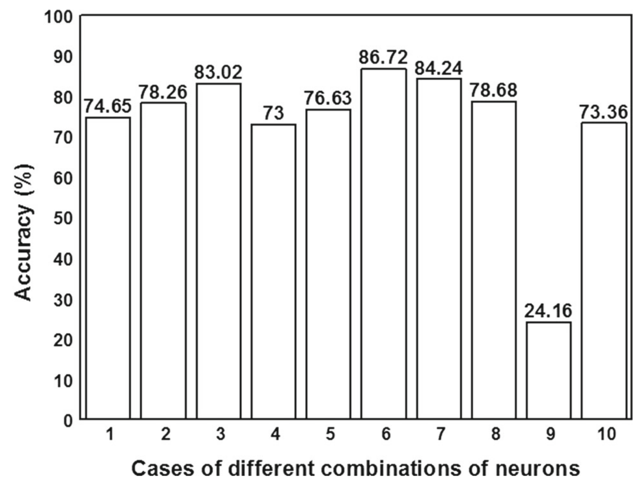


Fig. 27 Testing accuracy in eight layer model

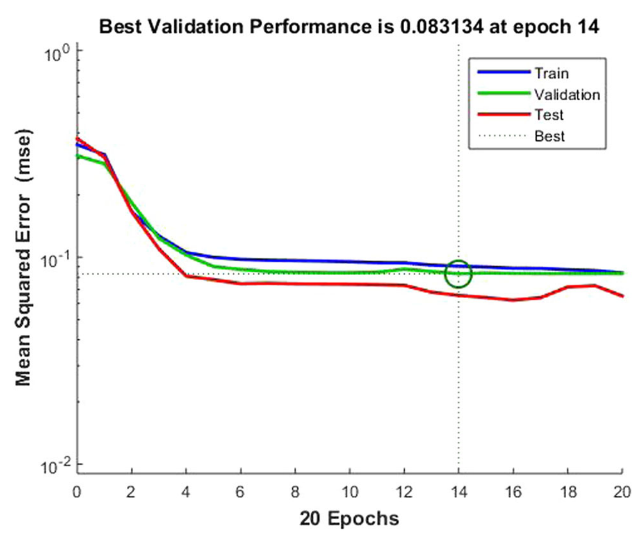


Fig. 28 Mean squared error of the best performance in eight layer model

Table 14 Attainment of evaluation metrics in three layer ANN architecture

Combination of neurons	Iterations	CPU time (in s)	Best validation performance	Accuracy (%)		Validation (%)	Average
				Training	Testing		
(10, 9)	14	7	0.069851	98.15	88.23	86.09	94.77
(11, 10)	5	<1	0.074415	86.40	85.14	84.23	85.67
(9, 8)	16	1	0.049075	97.87	88.27	89.78	95.23
(10, 8)	13	10	0.082669	86.70	81.82	81.88	85.25
(11, 9)	15	1	0.06235	98.04	87.99	87.49	94.86
(9, 7)	44	24	0.037037	96.83	85.43	92.45	94.46
12, 11)	12	10	0.030213	97.24	87.84	93.73	95.27
(12, 10)	20	14	0.039482	99.59	88.78	91.99	96.83
(8, 9)	21	13	0.040924	99.20	**90.19**	91.83	96.74
(7, 8)	11	9	0.04861	90.25	83.71	89.97	89.17
			Average	95.03	86.74	88.94	92.83
			SD	5.16	2.61	3.90	4.42

Bold value signifies the highest testing accuracy for distinct iterations

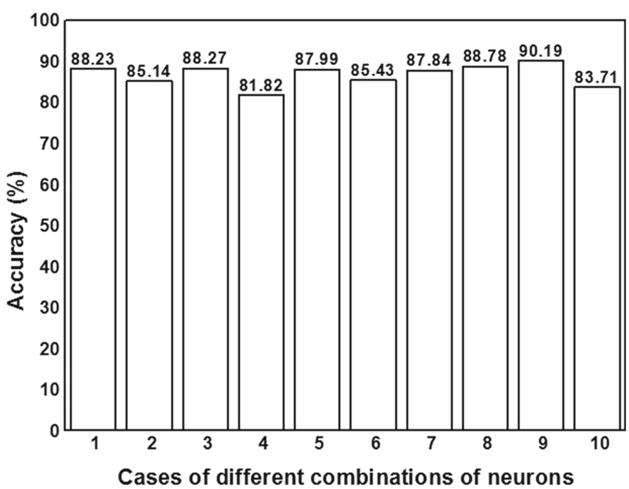


Fig. 29 Testing accuracy in three layer model

training purpose to test each of the kth subset. A single value obtained by taking the average of k-MSE demonstrates the performance of the method. The performance of the current proposal has been analyzed through 5-fold and 10-fold methods, presented in Table 22, to strongly support the claim about the correctness of the procedure.

7.3 Results of the proposed study on other datasets and comparison with the similar researches

The proposed study has also been implemented on Statlog dataset of 270 patients in which 189 and 81 patients are used for training and testing respectively and correctly identified the presence or absence of disease with an accuracy of 89.11%. The comparative results with the similar techniques are presented in Table 23.

Table 15 Attainment of evaluation metrics in four layer ANN architecture

Combination of neurons	Iterations	CPU time (in s)	Best validation performance	Accuracy (%)		Validation (%)	Average
				Training	Testing		
(10, 9, 8)	13	6	0.045159	96.56	86.43	90.63	94.12
(11, 10, 9)	19	11	0.050319	99.02	90.90	89.17	96.37
(9, 8, 7)	25	11	0.037002	97.22	90.56	92.59	95.53
(10, 8, 7)	18	7	0.020866	96.96	88.31	95.86	95.50
(11, 9, 8)	17	12	0.032278	97.55	93.24	93.63	96.29
(10, 9, 7)	8	7	0.070435	88.41	75.53	85.10	86.01
(7, 8, 6)	17	12	0.0089698	98.84	85.89	98.22	96.79
(9, 8, 6)	12	9	0.028079	96.38	88.19	94.48	94.85
(12, 11, 9)	27	18	0.03622	98.93	**98.18**	92.55	97.86
(10, 11, 9)	11	6	0.072017	93.71	92.37	84.42	92.13
			Average	96.36	88.96	91.66	94.55
			SD	3.21	5.95	4.43	3.39

Bold value signifies the highest testing accuracy for distinct iterations

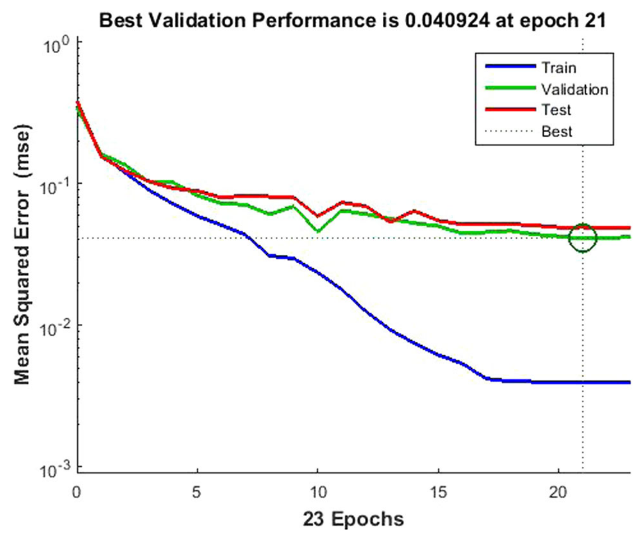


Fig. 30 Mean squared error of the best performance in three layer model

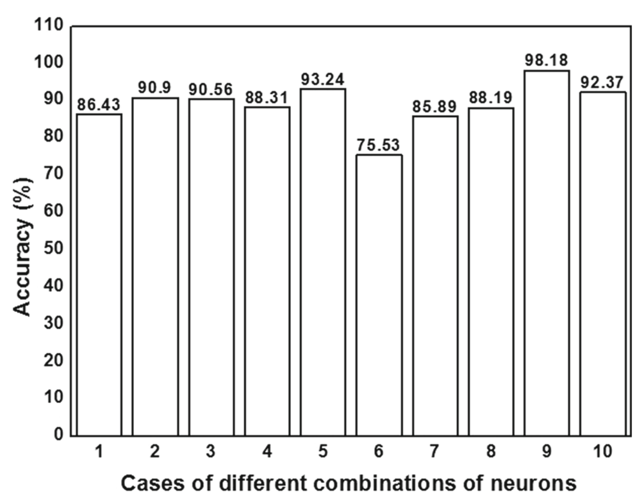


Fig. 31 Testing accuracy in four layer model

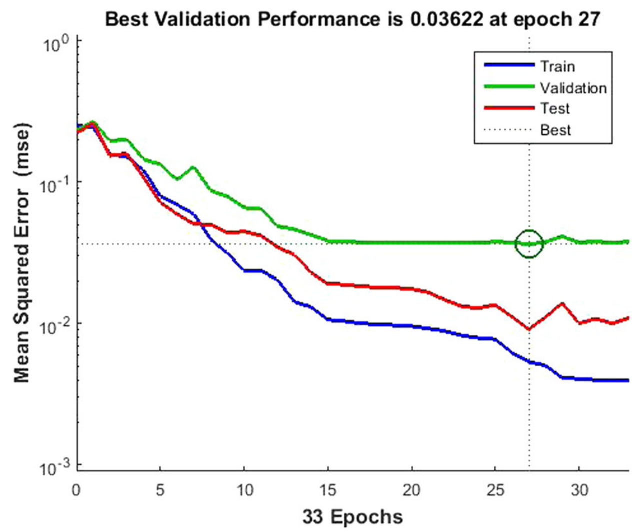


Fig. 32 Mean squared error of the best performance in four layer model

Table 16 Attainment of evaluation metrics in five layer ANN architecture

Combination of neurons	Iterations	CPU time (in s)	Best validation performance	Accuracy (%)		Validation (%)	Average
				Training	Testing		
(11, 10, 9, 8)	21	1	0.029936	99.75	92.51	94.04	97.82
(10, 9, 8, 7)	126	4	0.0090866	99.20	94.44	98.16	98.32
(9, 8, 7, 6)	21	<1	0.042487	99.17	91.74	91.36	96.89
(10, 8, 6, 5)	14	<1	0.044038	93.31	**94.76**	91.02	93.16
(10, 7, 6, 5)	20	1	0.059421	99.60	94.45	88.11	97.08
(10, 8, 7, 6)	16	<1	0.041029	97.61	92.53	91.75	95.98
(7, 9, 8, 6)	35	1	0.072288	98.86	86.18	85.20	94.90
(12, 10, 8, 7)	14	<1	0.078133	97.35	88.29	84.18	93.96
(9, 10, 7, 8)	20	<1	0.037016	97.15	89.13	92.41	95.22
(10, 7, 6, 5)	20	1	0.046362	99.20	93.19	90.31	97.00
			Average	98.12	91.72	90.65	96.03
			SD	1.94	2.92	4.09	1.69

Bold value signifies the highest testing accuracy for distinct iterations

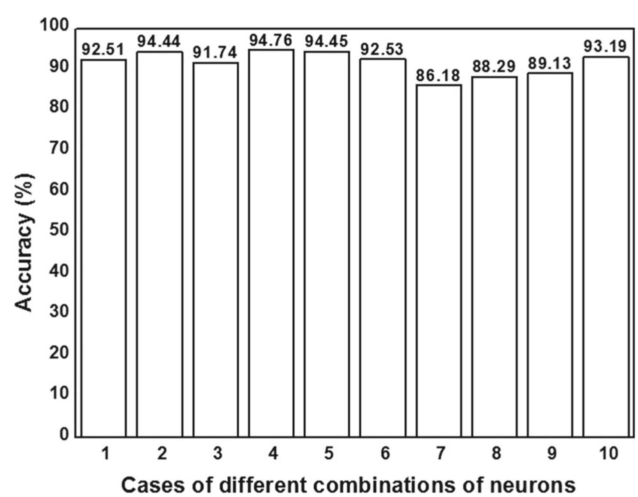


Fig. 33 Testing accuracy in five layer model

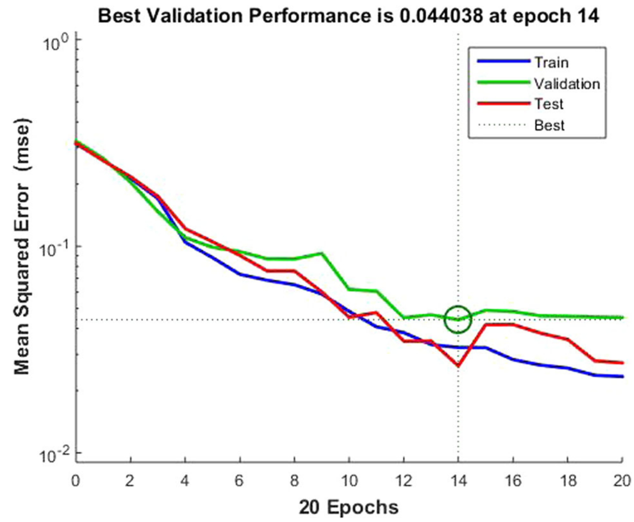


Fig. 34 Mean squared error of the best performance in five layer model

Table 17 Attainment of evaluation metrics in six layer ANN architecture

Combination of neurons	Iterations	CPU time (in s)	Best validation performance	Accuracy (%)		Validation (%)	Average
				Training	Testing		
(10, 9, 8, 6, 5)	44	1	0.021406	99.06	**97.60**	95.58	98.33
(9, 8, 6, 5, 4)	61	1	0.034005	96.53	86.81	92.71	94.54
(9, 8, 7, 6, 6)	22	<1	0.035317	95.61	92.15	92.66	94.64
(9, 7, 6, 5, 4)	32	1	0.055065	96.15	90.91	88.56	94.24
(10, 8, 7, 5, 4)	19	<1	0.018161	98.00	96.35	96.36	97.50
(9, 8, 6, 5, 3)	11	<1	0.057062	91.53	91.78	87.87	91.01
(11, 9, 8, 10, 5)	43	2	0.0091188	99.99	95.94	98.18	99.11
(10, 11, 9, 8, 7)	10	<1	0.060497	93.29	88.37	87.13	91.60
(12, 10, 9, 8, 7)	11	1	0.060412	95.97	84.97	87.40	93.04
(10, 9, 9, 8, 8)	60	2	0.018518	1.00	94.43	96.28	98.60
			Average	86.71	91.93	92.27	95.26
			SD	30.22	4.25	4.25	2.96

Bold value signifies the highest testing accuracy for distinct iterations

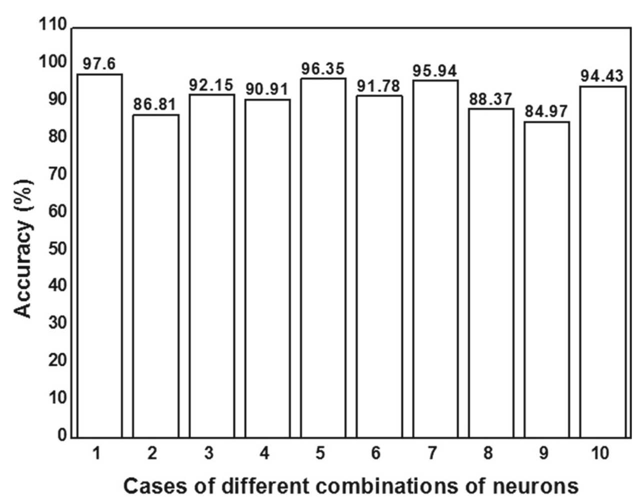


Fig. 35 Testing accuracy in six layer model

Fig. 36 Mean squared error of the best performance in six layer model

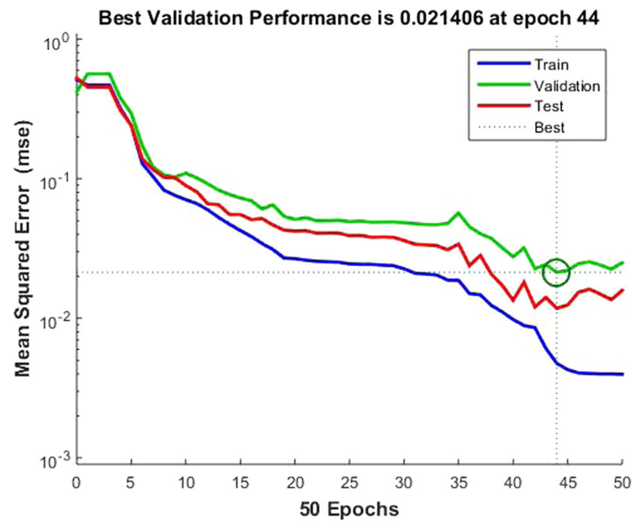


Table 18 Attainment of evaluation metrics in seven layer ANN architecture

Combination of neuron	Iterations	CPU time (in s)	Best validation performance	Accuracy (%)		Validation (%)	Average
				Training	Testing		
(10, 9, 8, 6, 5, 4)	23	1	0.019145(23)	98.80	92.76	96.07	97.50
(9, 8, 6, 5, 4, 7)	26	1	0.050312(26)	98.74	94.11	89.81	96.73
(9, 7, 8, 10, 6, 5)	37	14	0.027839(37)	97.247	**99.922**	94.137	97.2
(9, 7, 6, 5, 4, 6)	8	<1	0.066828(8)	89.28	81.53	85.68	87.55
(9, 8, 7, 5, 4, 7)	76	2	0.062778(76)	91.83	85.67	86.64	90.13
(9, 8, 6, 5, 3, 5)	23	1	0.036124(23)	92.88	90.68	92.49	92.48
(9, 8, 7, 6, 6, 5)	51	2	0.07885(51)	94.47	90.93	92.73	93.69
(10, 11, 9, 8, 7, 10)	24	1	0.027217(24)	98.02	96.19	94.55	97.22
(12, 10, 9, 8, 7, 11)	18	1	0.032056(18)	98.76	89.63	93.33	96.61
(7, 9, 6, 9, 8, 5)	18	1	0.065774(18)	97.95	85.76	86.45	94.36
			Average	95.80	90.72	91.19	94.35
			SD	3.44	5.44	3.77	3.41

Bold value signifies the highest testing accuracy for distinct iterations

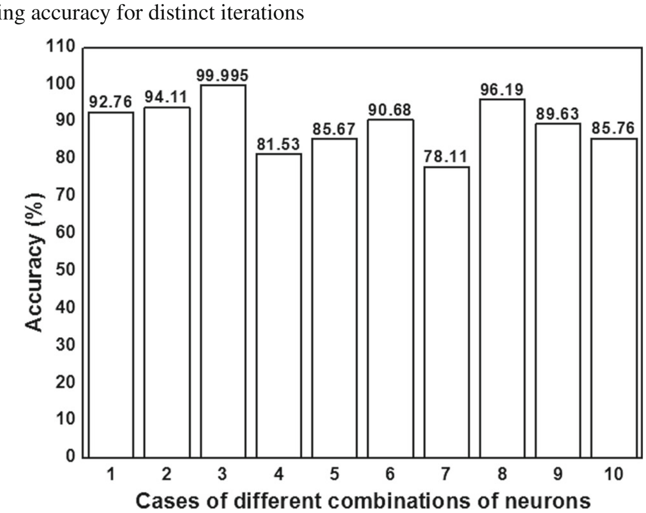


Fig. 37 Testing accuracy in seven layer model

Fig. 38 Mean squared error of the best performance in seven layer model

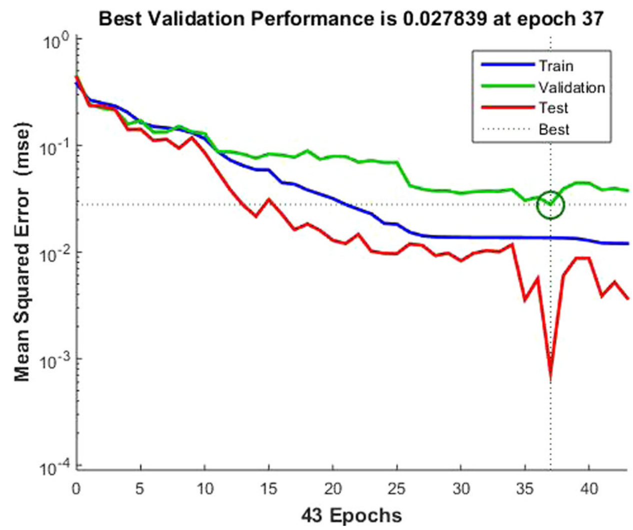
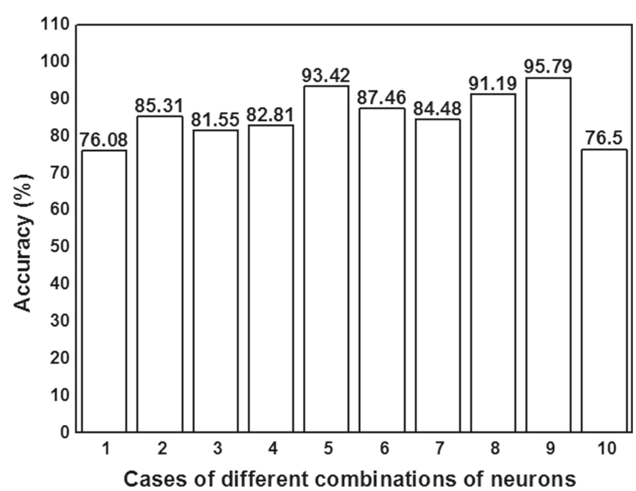


Table 19 Attainment of evaluation metrics in eight layer ANN architecture

Combination of neurons	Iterations	CPU time (in s)	Best validation performance	Accuracy (%)		Validation (%)	Average
				Training	Testing		
(10, 9, 8, 6, 5, 4, 3)	26	1	0.082977	90.68	76.08	81.77	87.14
(9, 8, 6, 5, 4, 7, 3)	17	<1	0.027785	94.49	85.31	94.45	93.04
(9, 8, 7, 6, 6, 5, 3)	9	1	0.062348	90.21	81.55	86.67	88.38
(9, 7, 6, 5, 4, 6, 2)	20	<1	0.070947	90.08	82.81	84.76	88.15
(9, 8, 7, 5, 4, 7, 3)	20	1	0.044096	94.47	93.42	90.80	93.79
(9, 8, 6, 5, 3, 5, 2)	20	<1	0.052208	92.09	87.46	89.19	90.89
(11, 9, 8, 10, 5, 7, 4)	33	2	0.056758	94.34	84.48	88.10	91.93
(10, 11, 9, 8, 7, 10, 5)	46	3	0.032568	99.24	91.19	93.32	97.14
(12, 10, 9, 8, 7, 11, 5)	19	1	0.064512	93.78	**95.79**	86.46	93.01
(7, 9, 6, 9, 8, 5, 4)	17	1	0.065215	90.50	76.50	85.56	87.71
			Average	92.99	85.46	88.11	91.12
			SD	2.88	6.66	3.92	3.25

Bold value signifies the highest testing accuracy for distinct iterations

Fig. 39 Testing accuracy in eight layer model



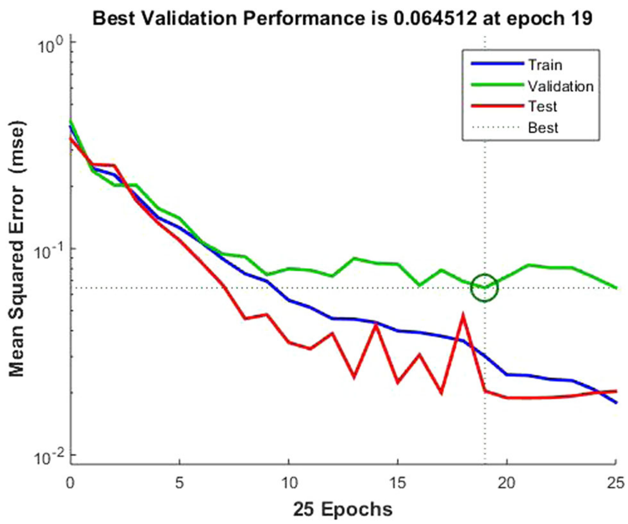


Fig. 40 Mean squared error of the best performance in eight layer model

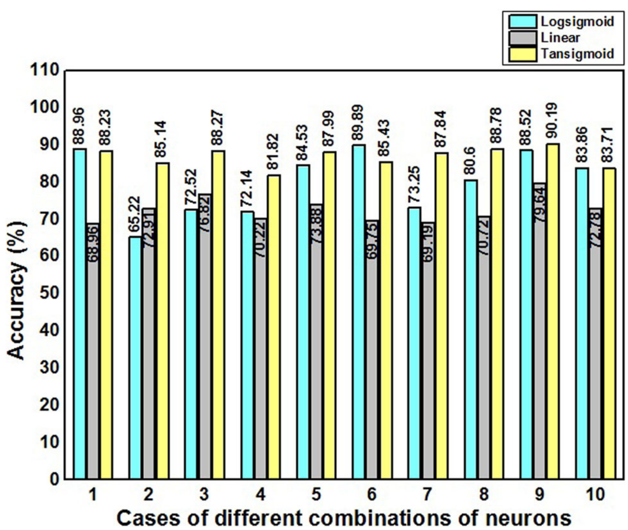


Fig. 41 Comparative analysis of three layer ANN architecture

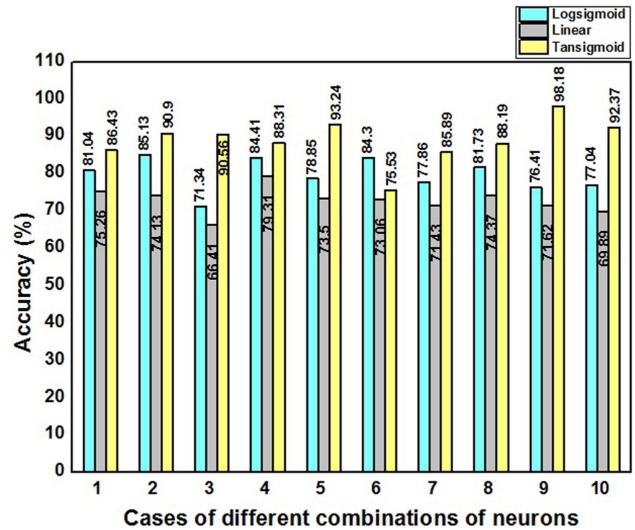


Fig. 42 Comparative analysis of four layer ANN architecture

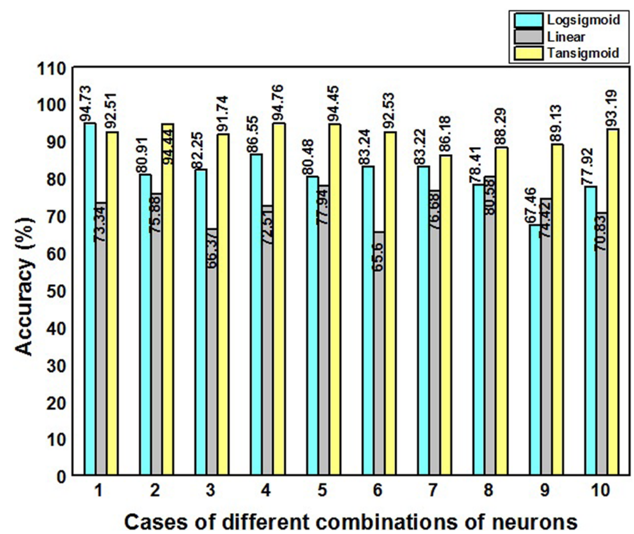


Fig. 43 Comparative analysis of five layer ANN architecture

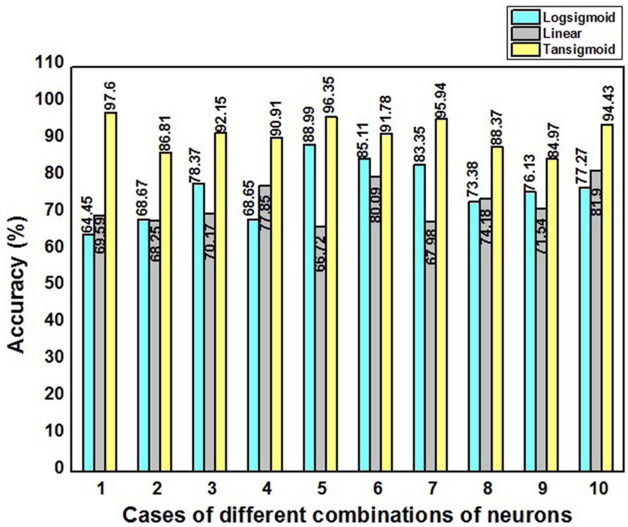


Fig. 44 Comparative analysis of six layer ANN architecture

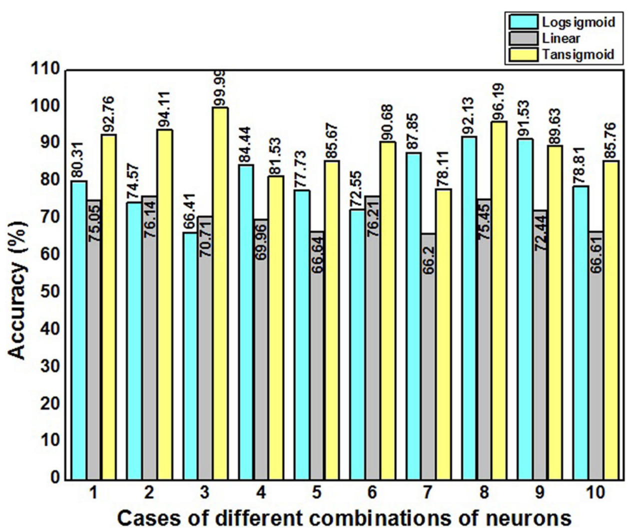


Fig. 45 Comparative analysis of seven layer ANN architecture

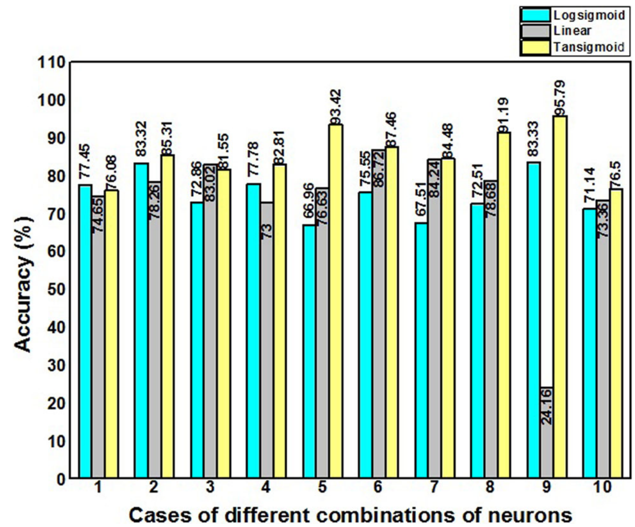


Fig. 46 Comparative analysis of eight layer ANN architecture

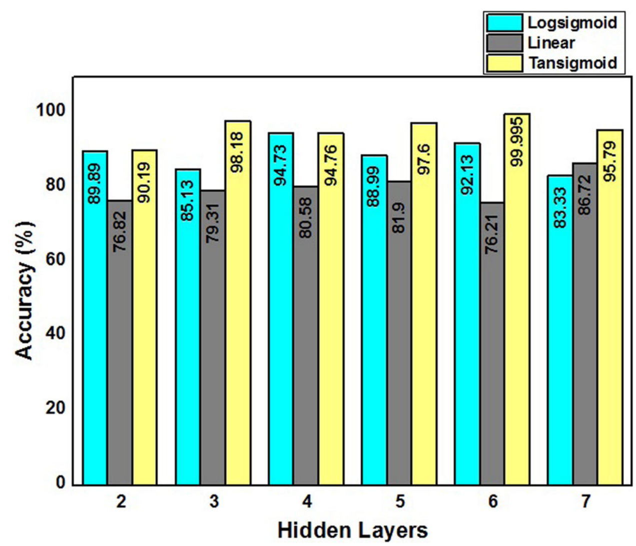


Fig. 47 Hidden layers versus accuracy

Table 20 Confusion matrix

Result of the diagnostic test		Physician diagnosis	
		Actual positive (1)	Actual negative (0)
Classifier result	Predicted positive (1)	147	17
	Predicted negative (0)	137	7

Table 21 The Outcomes of the proposed model for diagnosis of the heart disease

Specificity	Sensitivity	Accuracy	Precision	F-score	AUC
88.96%	95.45%	92.20%	0.89	0.92	0.90

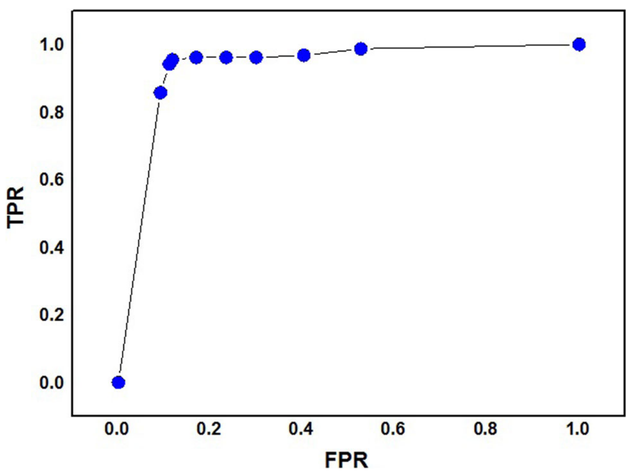


Fig. 48 Receiver operating characteristic curve

Table 22 Cross-validation

	Cross validation	
	Average MSE	Average accuracy (%)
5-fold	0.0817	89.20
10-fold	0.0742	90.99

8 Conclusion and future scope

The aim of this experimental work is to develop an ANN-Back Propagation based diagnostic model for the prediction of heart disease through comparison study of hidden layers and transfer functions. The best performance is yielded by *tan-hyperbolic* transfer function in each layer with an accuracy of 99.92% in seven-layer (six hidden layers) model of ANN. The proposed study attained an accuracy of 99.92% using Kaggle heart disease set. The performance was also validated through its test on Statlog dataset. The consistency of the system has been measured in terms of specificity (88.96%), sensitivity (95.45%), accuracy (92.20%), precision (0.89), f-score (0.92), AUC (0.90) and *k*-fold cross validation resulting in an overall accuracy of 92.88%. Further increase in the number of hidden layers reduces the accuracy of the proposed system. The results have been validated through the evaluation metrics and justified by computing statistics such as the values of accuracy, validation performance, average, variance and MSE. The results achieved during the proposed technique are highly accurate, encouraging researchers/ scholars to further explore mass adoption of this approach in the benefit of early and cost-effective heart disease diagnosis. In future, model can be tested on large scale datasets using SVM or other latest ML techniques.

Table 23 Comparison of the proposed study with the similar method

References	Input parameters	No. of patients	Data set	Approach	Accuracy	Software	Evaluation parameters	Transfer function
[18]	10	300	Sahara Hospital, Aurangabad	RBFNN		MATLAB		Gauusian
[19]	10	487	Central Clinical Hospital No. 2 of Russain Railways JSC	GA +MLPNN(Two buried 4-neuron layers)	93%	NeuroSolutions tool(5.0)		
[20]	13	303	UCI	GA+BPANN(one hidden layer with 7 neurons)	94.17%		Confusion Matrix	
[21]	13	899	UCI	MARS, RS, LR and ANN	MARS-ANN = 82.14%, LR-ANN = 78.57%, RS-ANN = 79.50%	RESE	RMSE, Type-1 error, Type-2 error	
[22]	8	303	CCF	BPANN+PSO	91.94%		Precision,Recall, $k = 10$ fold cross-validation	
[23]	13	297	CCF	GA+RFNN(one hidden layer with 7 neurons)	97.78%	Java	Sensitivity, specificity, precision, RMSE, PME. F-Score	
[24]	8	303	CCF	BPNN (3–12 hidden layers)	95.55%	MATLAB R2015a	Precision, Recall	Sigmoid
[25]	ECG signals	340	MIT-BIH	CNN (Total seven layers)	Near 99%	MATLAB R2015a	Confusion Matrix	Sigmoid
Ref	Input parameters	No. of patients	Data set	Approach	Accuracy	Software	Evaluation parameters	Transfer function
[26]	13	270	CCF and V. A. medical center	KNN, DT, ET, GB, RF, SVM, AB, NB, LR and BLPNN (one hidden layer with 6 neurons)	90.74%		AUC, ROC and PR Curve	Sigmoid

Table 23 continued

Ref	Input parameters	No. of patients	Data set	Approach	Accuracy	Software	Evaluation parameters	Transfer function
[27]	17	303	CCF	NB, Bayes Net, C4.5 DT, RF and MLPNN	NB = 84.16%, Bayes Net = 84.16%, C4.5 DT = 79.87%, RF = 80.53%, MLPNN = 81.52%	WEKA		
[28]	13	303 and 370	CCF and HIC	KNN, DT, ET, GB, RF, SVM, AB, NB, LR and ANN (one hidden layer with 20 neurons)	KNN = 87.11%, DT = 84.9%, ET = 94.41, GB = 93.36% SVM = 81.26%, AB = 83.01%, NB = 80.29%, LR = 80.77%, ANN = 86.72%	sci-kit learn (sklearn),pandas, matplotlib, seaborn and NumPyand	Confusion Matrix, Sensitivity, Specificity, F1-score, AUC, MCC, ROC curve and $k = 10$ cross-valdation	
[17]	13	70,000	Cardiovascular Disease dataset	LR, SVM, RF, KNN, NB, MLPNN and DT	LR = 73.6%, SVM = 74.1%, NB = 73.3%, KNN = 73.6%, MLPNN = 73.9%, DT = 74.8%	Python and Sci-kit learn	$k = 10$ fold cross-validation	
[29]	1499	15,000 ECG signals	Py-serial library	RNN and LSTM (32 hidden layers)		Arduino		Sigmoid
Proposed Method	13	1025	Kaggle heart dataset	BPNN (2–7 hidden layers)	99.92%	MATLAB R2015a	5-fold, 10-fold cross-validation	Sigmoid, Linear, Tanh

References

1. CDC Technical report (2019). https://www.cdc.gov/datastatistics/index.html
2. Virani SS, Alonso A, Benjamin EJ, Bittencourt MS, Callaway CW, Carson AP, Chamberlain AM, Chang AR, Cheng S, Delling FN (2020) Heart disease and stroke statistics 2020 update: a report from the American Heart Association. Circulation 141(9):139–596
3. Zhang Z (2018) Artificial neural network. In: Multivariate time series analysis in climate and environmental research, pp 1–35. Springer. https://doi.org/10.1016/B0-12-227410-5/00837-1
4. Abiodun OI, Jantan A, Omolara AE, Dada KV, Mohamed NA, Arshad H (2018) State-of-the-art in artificial neural network applications: a survey. Heliyon 4(11):00938. https://doi.org/10.1016/j.heliyon.2018.e00938
5. Mirbabaie M, Stieglitz S, Frick NR (2021) Artificial intelligence in disease diagnostics: a critical review and classification on the current state of research guiding future direction. Health Technol 11(4):693–731. https://doi.org/10.1007/s12553-021-00555-5
6. Rudomin P, Arbib MA, Cervantes-Pérez F, Romo R (2012) Neuroscience: from neural networks to artificial intelligence: proceedings of a US–Mexico Seminar Held in the City of Xalapa in the State of Veracruz on December 9–11, 1991 vol. 4. Springer. https://doi.org/10.1007/978-1-4612-2834-9
7. Yadav AK, Chandel SS (2014) Solar radiation prediction using artificial neural network techniques: a review. Renew Sustain Energy Rev 33:772–781. https://doi.org/10.1016/j.rser.2013.08.055
8. Costantini S, De Gasperis G, Olivieri R (2019) Digital forensics and investigations meet artificial intelligence. Ann Math Artif Intell 86(1):193–229. https://doi.org/10.1007/s10472-019-09632-y
9. Khanagar SB, Al-Ehaideb A, Vishwanathaiah S, Maganur PC, Patil S, Naik S, Baeshen HA, Sarode SS (2021) Scope and performance of artificial intelligence technology in orthodontic diagnosis, treatment planning, and clinical decision-making - a systematic review. J Dental Sci 16(1):482–492. https://doi.org/10.1016/j.jds.2020.05.022
10. Tumpa PP, Kabir MA (2021) An artificial neural network based detection and classification of melanoma skin cancer using hybrid texture features. Sens Int 2:100128. https://doi.org/10.1016/j.sintl.2021.100128
11. Kapoor R, Walters SP, Al-Aswad LA (2019) The current state of artificial intelligence in ophthalmology. Surv Ophthalmol 64(2):233–240. https://doi.org/10.1016/j.survophthal.2018.09.002
12. Bagi Ks, Shreedhara KS (2014) Biometric measurement and classification of IUGR using neural networks. In: 2014 international conference on contemporary computing and informatics (IC3I), pp 157–161. https://doi.org/10.1109/IC3I.2014.7019613
13. Poor NG, West NC, Sreepada RS, Murthy S, Görges M (2021) An artificial neural network-based pediatric mortality risk score: development and performance evaluation using data from a large north american registry. JMIR Med Inform 9(8):24079. https://doi.org/10.2196/24079
14. Juan-hua S, He-jun L, Qi-ming D, Ping L, Bu-xi K (2005) Prediction and analysis of the aging properties of rapidly solidified Cu–Cr–Sn–Zn alloy through neural network. J Mater Eng Perform 14(3):363–366. https://doi.org/10.1361/10599490524002
15. (2021) Machine learning applications in radiation oncology. Phys Imaging Rad Oncol 19:13–24. https://doi.org/10.1016/j.phro.2021.05.007
16. Liu C, Xie L, Kong W, Lu X, Zhang D, Wu M, Zhang L, Yang B (2019) Prediction of suspicious thyroid nodule using artificial neural network based on radiofrequency ultrasound and conventional ultrasound: a preliminary study. Ultrasonics 99:105951. https://doi.org/10.1016/j.ultras.2019.105951
17. Shorewala V (2021) Early detection of coronary heart disease using ensemble techniques. Inform Med Unlock 26:100655. https://doi.org/10.1016/j.imu.2021.100655
18. Khourdifi Y, Bahaj M (2019) Heart disease prediction and classification using machine learning algorithms optimized by particle swarm optimization and ant colony optimization. Int J Intel Eng Syst 12(1):242–252. https://doi.org/10.22266/ijies2019.0228.24
19. Atkov OY, Gorokhova SG, Sboev AG, Generozov EV, Muraseyeva EV, Moroshkina SY, Cherniy NN (2012) Coronary heart disease diagnosis by artificial neural networks including genetic polymorphisms and clinical parameters. J Cardiol 59(2):190–194. https://doi.org/10.1016/j.jjcc.2011.11.005
20. Amma NGB (2012) Cardiovascular disease prediction system using genetic algorithm and neural network. In: 2012 international conference on computing, communication and applications, pp 1–5. https://doi.org/10.1109/ICCCA.2012.6179185
21. Shao YE, Hou C-D, Chiu C-C (2014) Hybrid intelligent modeling schemes for heart disease classification. Appl Soft Comput 14:47–52. https://doi.org/10.1016/j.asoc.2013.09.020
22. Feshki MG, Shijani OS (2016) Improving the heart disease diagnosis by evolutionary algorithm of PSO and feed forward neural network. In: 2016 artificial intelligence and robotics (IRANOPEN), pp 48–53. https://doi.org/10.1109/RIOS.2016.7529489
23. Uyar K, Alhan A (2017) Diagnosis of heart disease using genetic algorithm based trained recurrent fuzzy neural networks. Proc Comput Sci 120:588–593. https://doi.org/10.1016/j.procs.2017.11.283
24. Karayılan T, Kılıç Ö (2017) Prediction of heart disease using neural network. In: 2017 international conference on computer science and engineering (UBMK), pp 719–723. IEEE
25. Gawande N, Barhatte A (2017) Heart diseases classification using convolutional neural network. In: 2017 2nd international conference on communication and electronics systems (ICCES), pp 17–20. IEEE
26. Costa W, Figueiredo L, Alves E (2019) Application of an artificial neural network for heart disease diagnosis. In: XXVI Brazilian Congress on Biomedical Engineering, pp 753–758 . Springer. https://doi.org/10.1007/978-981-13-2517-5_115
27. Latha CBC, Jeeva SC (2019) Improving the accuracy of prediction of heart disease risk based on ensemble classification techniques. Inform Med Unlock 16:100203. https://doi.org/10.1016/j.imu.2019.100203
28. Muhammad Y, Tahir M, Hayat M, Chong KT (2020) Early and accurate detection and diagnosis of heart disease using intelligent computational model. Sci Rep 10(1):1–17. https://doi.org/10.1038/s41598-020-76635-9
29. Shihab AN, Mokarrama MJ, Karim R, Khatun S, Arefin MS (2020) An IoT-based heart disease detection system using RNN. In: International conference on image processing and capsule networks, pp 535–545. Springer
30. Tiwari S (2020) Activation functions in neural networks. https://www.geeksforgeeks.org/
31. Duggal R, Gupta A (2017) P-telu: parametric tan hyperbolic linear unit activation for deep neural networks. In: Proceedings of the IEEE international conference on computer vision workshops, pp 974–978. https://doi.org/10.1109/ICCVW.2017.119
32. Wallach D, Goffinet B (1989) Mean squared error of prediction as a criterion for evaluating and comparing system models. Ecol Model 44(3):299–306. https://doi.org/10.1016/0304-3800(89)90035-5
33. Ting KM (2017) In: Sammut C, Webb GI (eds) Confusion matrix. Springer, Boston. https://doi.org/10.1007/978-1-4899-7687-1_50
34. Davis J, Goadrich M (2006) The relationship between precision-recall and roc curves. In: Proceedings of the 23rd international conference on machine learning, pp 233–240

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.